Commodore WORLD

Nº 10 - DICIEMBRE

315 PTAS.

Concursos
¡¡¡Bueeení...simos...!!!





¡Hazte socio del Club de Soft de España!

Tendrás multitud de ventajas...

- Recibirás puntual información de las novedades de soft que haya en el mercado.
  Te enviaremos un catálogo bimensual que incluye más de 200 programas para Commodore 64 y Spectrum.
- Podrás adquirir programas con un 30% de descuento, por ser miembro del CLUB DE SOFT DE ESPAÑA.
- Obtendrás descuentos de hasta un 20% sobre hardware.

¿COMO PUEDES HACERTE SOCIO DEL CLUB DE SOFT DE ESPAÑA?

- Sólo tendrás que abonar 1.000 Ptas. como cuota UNICA de inscripción.
- Como regalo de bienvenida a nuestro CLUB de SOFT de ESPAÑA podrás elegir entre dos programas:

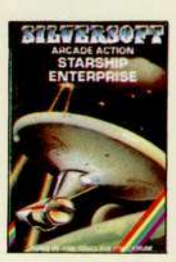
AJEDREZ	STARSHIP ENTERPRISE
SOLO FLIGHT	SLIPPERY SID
para CBM-64	para SPRECTRUM

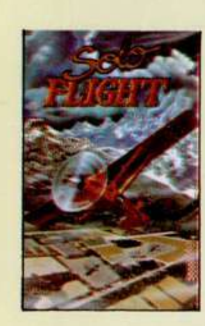
Te enviaremos el programa elegido completamente GRATIS, junto con la confirmación de tu pertenencia al CLUB de SOFT de ESPAÑA.





CLUB de SOFT de ESPAÑA





DIFUSION POR AMISTAD

Habla a tus amigos de las enormes ventajas que supone pertenecer al CLUB DE SOFT DE ESPAÑA. Si junto con tu inscripción se inscriben dos amigos tuyos por tu recomendación, te **regalamos** un programa de juegos.



CUPON DE INSCRIPCION AL CLUB DE SOFT DE ESPAÑA

Eraso, 12 - Tel. 24610 94 - 28028-MADRID

Nombre ____________________

Dirección ____________________

Ciudad ____________________ Tel. ____________

Deseo pertenecer al CLUB DE SOFT DE ESPAÑA. Ruego me envien como regalo de bienvenida al CLUB el programa ____________________

Firma

El CLUB DE SOFT DE ESPAÑA te enviará el programa solicitado, junto con un reembolso de 1.000 Ptas. más gastos de envío, como cuota de inscripción al Club.

DIFUSION POR AMISTAD

NOMBRE de un amigo ____________________

Dirección ____________________

Programa de regalo de bienvenida ____________________

Firma

DIFUSION POR AMISTAD

NOMBRE de un amigo ____________________

Dirección ____________________

Programa de regalo de bienvenida ____________________

Firma

Commodore WORLD

Commodore World está publicado por SIMSA y la colaboración de todos nuestros lectores

EQUIPO
Manuel AMADO; Juan L. AMESTI; Nieves CHESA; José Luis ERRAZQUIN; M.ª Jesús GARCIA; Miguel A. HERMOSELL; Alvaro IBAÑEZ; María LOPEZ; Juan MARQUEZ; Juan MARTINEZ; Pere MASATS; Victoria MORALES; Rafael PARDO; Diego ROMERO; Albert SANGLAS; Jordi SASTRE; Valerie SHANKS; Francisco ZABALA... ...Y NUESTROS LECTORES

SIMSA
Coordinador María López
c/Barquillo, 21-3º. Izda. - 28004
Madrid. Teléf.: (91) 231 23 88/95
DELEGACION EN BARCELONA:
Sant Gervasi de Cassoles, 39 despacho 4.
08022 Barcelona
Tels.: (93) 212 73 45/212 88 48

Colabora MEC-COMMODORE con Club Commodore
Coordinador Pere Masats
Valencia, 49-53
08015 Barcelona
Teléf.: (93) 325 50 08



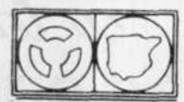
Imprime:
IBERDOS, S. A.
Germán Pérez Carrasco, 24.
28027 Madrid
Depósito Legal: M-2944-1984

SUMARIO

CONTENIDO — PAG.

PROXIMO NUMERO

RUTINAS BASIC (III): Paginación.
UNA SERIE DE UTILIDADES Y TRUCOS
CASTELLANIZACION DEL COMMODORE: ACENTOS Y EÑES
CONSTRUCTOR DE DATOS
EL ARTE DE REALIZAR ORGANIGRAMAS
PROGRAMA MATEMATICO
DESCUBRIR EL SECRETO DE LA ESTRUCTURA INTERIOR DEL VIC



Commodore World es miembro de CW Communications/Inc., el grupo editorial más grande y de mayor prestigio en el mundo en lo que se refiere al ámbito informático. Dicho grupo tiene a su cargo la edición de 52 publicaciones relacionadas con los ordenadores en 19 países. El grupo editorial está integrado por:
ALEMANIA: Computerwoche, Micro Computerwelt, PC Welt, Software Markt, CW Edition/Seminar, Computer Business, Run (Commodore). **ARABIA SAUDI:** Saudi Computerworld. **ARGENTINA:** Computerworld/Argentina. **AUSTRALIA:** Australasian Computerworld, Micro Computer Magazine, PC World. **BRASIL:** DataNews, Micro-Mundo. **DINAMARCA:** Computerworld/Danmark, Micro Verden. **ESPAÑA:** Computerworld/España. MicroSistemas, Commodore World. **ESTADOS UNIDOS:** Computerworld, Hot CoCo, InCider, InfoWorld, Microcomputing, PC World, 80-Micro, Mac World Micro Market Word, PC Jr. World Run (Commodore). **FINLANDIA:** Mikro. **FRANCIA:** Le Monde Informatique. **HOLANDA:** Computerworld Benelux, Micro/Info. **ITALIA:** Computerworld Italia. **JAPON:** Computerworld Japan, Perso Comworld, PC World. **MEXICO:** Computerworld/México, Compumundo. **NORUEGA:** Computerworld Norge, Mikro Data. **REINO UNIDO DE GRAN BRETAÑA:** Computer Management. Computer Business Europe. **REPUBLICA POPULAR DE CHINA:** China Computerworld. **SINGAPUR:** Asian Computerworld. **SUECIA:** ComputerSweden. MikroDatorn, Min Hemadator.

SUPER-INTERESANTISIMO
DE TODO UN POCO

NUMEROS ATRASADOS

Primero empezaremos con las cosas prácticas y luego pasaremos a asuntos más frívolos (pero no por eso menos interesantes). Como todas las cosas buenas de la vida tienen que terminar, así nos ocurrió con el número 1 de "Commodore World", y nos está ocurriendo con los otros ejemplares. Todavía disponemos de los primeros números pero algunos están en mal estado y aunque están utilizables tampoco se encuentran en la flor de la vida. Si queréis estos ejemplares nosotros seguiremos enviando pero el que avisa no es traidor.

Por otro lado, el número 1 está definitivamente agotado pero nos gustaría dar algún tipo de servicio a la gente que lo sigue pidiendo. De momento no hemos decidido si este servicio consistirá en unas fotocopias o cintas pero el boletín correspondiente aparecerá en la revista a su debido tiempo. Mientras tanto a esperar toca.

SIMO Y CAMISETAS "COMMODORE WORLD"

Nos ha alegrado mucho estar en el SIMO y saludar a los muchos suscriptores y lectores que nos han venido a ver. A muchos de ellos ya los conocíamos de tanto hablar por teléfono y escribirnos y la verdad es que esta feria nos ha brindado la ocasión de entrar en contacto directamente con los miembros de nuestra gran familia "commodoriana".

Los que habéis pasado por el stand os habréis fijado que algunos de nosotros íbamos vestidos a la última moda, con una prenda exclusivísima que nos quedaba la mar de bien. Estamos hablando de nuestra camiseta de "Commodore World". Nos hicimos el modelito con el fin de poder identificarnos mutuamente entre toda la masa de gente que acude al SIMO, y nos hemos quedado gratísimamente sorprendidos de la cantidad de gente que nos ha preguntado si se podía comprar una. Pues bien, la verdad es que no lo habíamos pensado, pero ya que realmente os apetece la idea, en el próximo número os daremos los detalles ya que en este no nos da tiempo.

MERCADILLO

Solamente recordaros que os atengáis a las regulaciones que se publicaron en el número noviembre (el 9), en pág. 4. Esto evitará que tengamos que rechazar anuncios.

CONCURSOS - CONCURSOS

ESCOLAR

Recordamos a todos que el último día de admisión de trabajos es el próximo *31 de diciembre*, leeros bien las bases.

EL DE SIEMPRE
—COLABORADORES—
PROGRAMAS Y MAGIA

El sorteo de todas las colaboraciones publicadas en los números de julio a diciembre (ambos incluidos) será, como siempre, público y tendrá lugar el próximo día 14 de diciembre, viernes, a las 17 horas en nuestra redacción, Barquillo, 21, estáis invitados.

A POR LOS 8.000...
Y A VIAJAR

Falta bastante menos, último número de suscriptor hoy, día 19 de noviembre, es el 6.112.

Muy Felices Navidades a todos, en nombre del Equipo

Convocatoria del Primer Concurso Nacional Escolar de Programación patrocinado por Commodore World y Microelectrónica y Control

BASES

1º Podrán participar en el mismo todos aquellos niños y jóvenes que estén en edad escolar pre-universitaria o equivalente.

2º Habrá tres grupos definidos:

a) E.G.B.

b) B.U.P.

c) F.P.

3º Los trabajos pueden ser presentados por equipos o por individuales.

4º Los trabajos presentados por equipos deben especificar:

a) Nombre del equipo.

b) Centro al que representan.

c) Nivel de enseñanza que cursan (E.G.B.-B.U.P-F.P.).

d) Nombre del profesor, monitor o coordinador.

5º Los trabajos presentados por individuales deben ir acompañados de:

a) Nombre del autor.

b) Edad y curso que estudia.

c) (¡) Si concursa representando a un centro: nombre del mismo y nombre del profesor responsable.

(¡¡) Si concursa por si mismo: nombre del colegio donde estudia y firma del director o profesor responsable avalando que en tanto cuanto conoce al concursante el trabajo que presenta es suyo propio.

6º El tema del concurso es la realización de un programa educativo sobre materia libre a elegir por los concursantes. La extensión del programa es también de libre elección.

7º Cada centro puede presentar el número de equipos o concursantes individuales que desee, siempre que cada trabajo sea enviado por separado.

8º Los trabajos deben incluir:

a) Nombre el programa.

b) Nombre del computador para el que está confeccionado y periféricos adicionales que se necesitan (si fuese el caso).

c) Objetivo del programa.

d) Explicación detallada del programa (a máquina o impresora).

e) Listado completo a impresora.

f) Grabación en cinta o disco.

9º Todos los trabajos presentados deben ser originales e inéditos.

10º Los centros que pueden presentar equipos o concursantes individuales son:

a) Todo colegio, escuela o instituto, público o privado que imparta las enseñanzas de E.G.B., B.U.P., o F.P.

b) Centros de Informática que tengan grupos de enseñanza específica para niños y jóvenes menores de 18 años.

c) Clubs de informática que mantengan actividades específicas para niños y jóvenes menores de 18 años.

11º Los trabajos deben ser enviados antes del 31 de diciembre de 1984 a SIMSA, c/ Barquillo, 21-3º Izqda. 28004 Madrid.

12º El fallo del concurso se dará a conocer en la primera semana de marzo de 1985 en el lugar, fecha y hora que se **indicarán**.

13º Los trabajos premiados quedarán propiedad de los patrocinadores.

RELACION DE PREMIOS EN PAG. 50

CONCURSOS PERMANENTES

Colaboraciones de Programas y Magia

La verdad sea dicha: os estáis portando; las colaboraciones que nos están llegando son numerosas y de muy buena calidad. Por lo tanto, vamos a aclarar las condiciones de nuestros concursos permanentes.

1) Los sorteos se realizarán en los meses de junio y diciembre.

2) Se sortearán siempre seis premios, tres de 15.000, 10.000 y 5.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las colaboraciones publicadas.

3) Se sortearán siempre seis premios, tres de 3.000, 2.000 y 1.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las contribuciones publicadas en la sección de "Magia".

4) Adicionalmente, se sortearán diversos premios de cintas, juegos, etc. donados por firmas y distribuciones, entre todas las colaboraciones.

5) Los autores de las colaboraciones vendidas dentro de nuestro "Servicio de Cintas" percibirán el 20% del precio de la cinta.

6) Todas las colaboraciones deben venir escritas a máquina y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestros lectores más jóvenes pueden escribir a mano pero con letra muy clara.

7) Quedan automáticamente descalificadas tanto del sorteo como del "Servicio de Cintas" las colaboraciones que hayan sido enviadas a otras revistas.

8) Las colaboraciones se enviarán a "Commodore World", *C/Barquillo, 21-3 izda. 28004 Madrid.*

MAS VIAJES Y OTRO REGALO DE "IMPRESION"... A POR EL SUSCRIPTOR 8.000

Es evidente que a nuestros sucriptores les gusta viajar. De momento Joan Pujiula visitará los Estados Unidos en julio y Juan Mejuto Iglesias irá a Inglaterra en el otoño. Queríamos sortear algo un poco especial ENTRE TODOS LOS SUCRIPTORES cuando lleguemos al suscriptor número 8.000, y hemos optado por dos premios, uno concedido por Microelectrónica y Control y otro concedido por "Commodore World".

MEC regalará una impresora MPS 801 (o similar), y nosotros hemos escogido un surtido de viajes muy "chulos" para dos personas para que el ganador elija el que más le apetezca. La selección que hemos hecho es la siguiente:

Viajes de 4 días en hoteles de primera clase a uno de estos lugares: París, Lisboa, Atenas, Amsterdam, Niza, Viena.

O viajes de una semana en hoteles de 4/5 estrellas: Canarias o Mallorca.

NO OLVIDEIS
NUESTRA NUEVA
DIRECCION:
BARQUILLO, 21-3º IZDA.
28004 MADRID
TELEF.: 231 23 88
231 23 95

TIRADA DE COMMODORE WORLD
Certificación de 28.000 ejemplares

Según certificado firmado por la auditoría internacional DELOITTE HASKINS SELLS, S. A. el 5 de marzo de 1984, la tirada de Commodore World es de 28.000 ejemplares. Este certificado se encuentra en las oficinas de S.I.M.S.A. para comprobación de cualquier persona o entidad que lo desee.

Convocatoria del primer concurso nacional de programación en 80 columnas de Commodore-64 y 40-80 columnas del VIC-20 patrocinado por Ferre-Moret, S.A. Con la colaboración de Commodore-World, más de 500.000 ptas. en premios

BASES

1º Podrán participar todos los usuarios de COMMODORE-64 y VIC-20 que hayan obtenido la tarjeta de 80 columnas de FERRE-MORET, S.A.

2º Cada concursante podrá presentar, un programa realizado por él o por un equipo del que deberán indicar un titular.

3º Se enviará la explicación detallada del programa redactado en español, escrito a máquina. El programa se enviará en cassette o disco, adjuntando su correspondiente listado del programa.

4º Se enviarán por correo certificado a la siguiente dirección, FERRE-MORET, S.A. calle Buenos Aires nº 30, 2º-3.ª de BARCELONA-36. Antes del 31 de enero de 1985.

5º Los programas no premiados podrán ser solicitados por sus propietarios en el plazo de treinta dias después del fallo.

6º El resultado se comunicará, directamente a los ganadores y se publicará en la revista COMMODORE WORLD.

7º Los premios se entregarán durante la primera semana de marzo de 1985 coincidiendo con la entrega de premios del PRIMER CONCURSO NACIONAL ESCOLAR DE PROGRAMACION.

8º Los programas premiados quedarán en propiedad de FERRE-MORET, S.A.

9º Los programas que no obtengan premio, recibirán, en caso que así lo considere FERRE-MORET, S.A., ofertas de compra de sus programas.

10º Los premios ascenderán a 500.000 Ptas. repartidos de la siguiente manera:

1º 250.000 Ptas.
2º 150.000 Ptas.
3º 100.000 Ptas.

11º Todos los participantes, que no hayan recibido premio u oferta de compra, podrán participar en el concurso de colaboradores que habitualmente COMMODORE WORLD viene realizando con un premio extra de FERRE-MORET, S.A., de 50.000 Ptas.

12º Los trabajos deberán de ser inéditos y desarrollar temas de utilidad, programas profesionales, científicos, educativos, etcétera.

FERRE-MORET, S.A. se reserva el derecho de modificar las bases de este certamen, dejarlo desierto o suprimirlo, por causa de fuerza mayor. En este caso la empresa se compromete a comunicarlo a través de COMMODORE WORLD.

Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)

[CRSRU] = Tecla cursor hacia arriba con SHIFT)

[CRSRR] = Tecla cursor a la derecha (sin SHIFT)

[CRSRL] = Tecla cursor a la izquierda (con SHIFT)

[HOME] = Tecla CLR/HOME (sin SHIFT)

[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora. ■

INDICE DE ANUNCIANTES

Por Gary McCLELLAN (RUN. EE.UU.)
Traducido por Valerie SHANKS y adaptado por Diego ROMERO

Un sencillo procesador de textos para el C-64

El programa fue escrito para un Apple, pero funciona perfectamente —con unas pocas modificaciones— en el C-64. El artículo y el listado presentados a continuación describen las modificaciones y las adiciones que se llevaron a cabo sobre el programa original. El programa funciona con una impresora y unidad de discos Commodore.

El programa modificado se parece en lo más básico al programa original. El texto se escribe directamente del teclado y se almacena en elementos de "strings" de caracteres en un "array" de "strings". Cada elemento tiene asignado un número de referencia, y cada uno puede almacenar hasta 240 caracteres. Estos bloques de texto pueden ser editados y revisados, grabados en disco o leídos del mismo, y volcados a impresora.

En el programa original los comandos VTAB y HTAB de Apple se utilizaban con frecuencia. Dado que el C-64 no tiene el equivalente del comando VTAB, se han utilizado sentencias PRINT con comandos de la tecla del cursor para generar los formatos de pantalla. Los valores de las teclas del cursor, de RVS ON y RVS OFF y la tecla CLR/HOME se asignan a las variables de "strings" en las líneas 15 y 17 del programa y son los que se utilizan, en vez de los caracteres gráficos de Commodore, en este listado. Se hizo de esta forma para que el listado fuese más fácil de leer.

Se han modificado los caracteres de control utilizados para escribir un texto nuevo. Al pulsar la tecla DEL se borra el último espacio o carácter escrito. Para introducir los caracteres de control al empezar un bloque de texto, se utilizan caracteres de control de una sola tecla, en vez de usar una secuencia de tecla de control/tecla secundaria.

El símbolo de flecha arriba inicia un nuevo párrafo de texto con una tabula-

C-64



ción de cinco espacios. El símbolo de flecha hacia la izquierda inicia el texto en la siguiente línea del margen izquierdo.

Pulsando la tecla del signo de más (+) se salta una línea y el texto sigue a continuación. Al volcar el texto a impresora el signo "@" lo imprime con caracteres de doble anchura. Al igual que en el programa original, al pulsar la tecla Return dos veces se vuelve al menú.

Cuando se elige la función de escribir, el programa avisa al usuario que puede empezar. Pulsando la tecla Return se elige el primer bloque de texto vacío que quede disponible. Si el usuario elige un bloque que contiene un texto, aparecerá un mensaje de "ref ocupada" y se buscará el siguiente bloque vacío que quede disponible.

Edición ampliada

La función de revisar/editar también ha sido ampliamente modificada para no tener que volver a escribir encima del texto. Después de seleccionar la opción revisar/editar en el menú, pulsa la tecla E para que el programa entre en la modalidad de edición. El bloque de texto aparecerá junto con los comandos de edición en la parte inferior de la pantalla. Ahora el texto puede ser tecleado o modificado.

Si pulsas la tecla CLR/HOME el cursor se colocará encima del primer carácter del bloque de texto. Pulsando la tecla CLR/HOME con SHIFT se coloca el cursor encima del último carácter del bloque de texto. La tecla de cursor derecha/izquierda desplaza el cursor en cualquiera de estas dos direcciones. La tecla INST/DEL borra el carácter situado a la izquierda del cursor y desplaza el resto del texto un espacio hacia la izquierda. La tecla INST/DEL con SHIFT desplaza el carácter donde se encuentra el cursor junto con todo el texto a la derecha del cursor, un espacio a la derecha, e intercala un espacio en blanco en el lugar donde se encuentra el cursor. La tecla Return guarda todo el texto en pantalla, sea cual sea la posición del cursor, y el bloque de texto volverá a aparecer en pantalla en la modalidad de revisar.

Cuando se edita un texto con el bloque casi lleno, es posible que no se vean los desplazamientos de texto al introducir y borrar pulsando las teclas. Sin embargo, estas funciones se están realizando, y al pulsar la tecla Return, se puede ver el bloque de texto en pantalla de nuevo, en la opción de revisar incluyendo todas las modificaciones realizadas. Al pulsar la tecla E, la pantalla vuelve a la opción de edición, de modo que se puede continuar escribiendo.

La opción insertar/copiar permite la introducción de un bloque de texto donde sea necesario, sin destruir el texto existente.

La rutina de salvar en disco guarda todo el texto en un fichero secuencial llamado TEXTFILE. El fichero nuevo se salva encima de un fichero guardado previamente en el disco, de modo que hay que tener cuidado al guardar el fichero.



Dado que el comando Input del C-64 utiliza 80 caracteres, la rutina que carga el fichero de disco utiliza el comando Get y construye cada elemento del "array" a medida que va avanzando. Esto resulta bastante lento, de modo que a medida que cada elemento se construye, aparece un punto en pantalla a continuación del mensaje CARGANDO ARCHIVO DE TEXTO para que el usuario se dé cuenta de que se está cargando el fichero.

Después de seleccionar la opción revisar/editar en el menú, pulsa la tecla E para que el programa entre en la modalidad de edición.

Adiciones

Una opción de insertar/copiar fue incorporada al menú con el fin de poder copiar un bloque de texto de un número de referencia determinado a otro o insertar un bloque de texto vacío en cualquier número de referencia. Una vez hecha la selección, el programa te indicará que elijas o bien la función de copiar o bien la función de insertar.

Si eliges la función de copiar, el programa te preguntará el número del bloque de texto que será copiado y el número del bloque donde se tiene que copiar, y a continuación te preguntará si estas seguro. Tienes que contestar que sí antes de que se pueda iniciar la función. Si eliges la función de insertar, el programa te indica que teclees el número de referencia deseado. La rutina desplazará los bloques de texto existentes hacia un número de referencia más alto, empezando por la referencia del bloque seleccionado. Esta facilidad te permite introducir bloques de texto cuando sea necesario, sin tener que destruir el texto existente.

La rutina para formatear la impresora constituye una adición muy importante. Cuando se selecciona la opción de copia impresa, el programa te indica que elijas la rutina de formatación o los parámetros por omisión estandard.

Los valores por omisión son los siguientes: margen izquierdo de cinco caracteres, longitud de línea de 70 caracteres (incluyendo los caracteres del margen izquierdo), margen superior de tres líneas y margen inferior de tres líneas. A medida que se va imprimiendo el texto, el programa busca un espacio entre los últimos diez caracteres de la línea y, si lo encuentra, envía un comando de retorno de carro a la impresora. Si no, la línea termina cuando el contador de caracteres llega a la máxima longitud de la línea. Como resultado de esto el margen derecho queda desigual al no dividir una palabra (a no ser que encuentre una palabra de más de diez caracteres). Esto se hace así principalmente para que el texto impreso resulte más fácil de leer.

El programa también pide el principio de página para que pueda controlar el número total de las líneas impresas en una página y saltar a la página siguiente (dejando los margenes superior e inferior) para evitar que se imprima en el pliegue de la página. Tienes que alinear el principio de página con la cinta de la impresora.

Una lista de todas las variables utilizadas empieza en la línea 5000 con el fin de facilitar cualquier modificación adicional al programa. Se inician los valores por omisión de formato de la impresora en la línea 10 y estos se pueden modificar fácilmente con el fin de acoplarse a una necesidad determinada.

Suma de Control del procesador de texto

```
2 POKE53281,1:POKE53280,12:PRINT"[BLK]
"
10 LM=5:LL=70:PS=3:PE=3:PP=1:RE=500:
REM PARAMETROS DE IMPRESION POR DEFE
CTO
15 RO$=CHR$(18):RF$=CHR$(146):CL$=CH
R$(157):CR$=CHR$(29):CD$=CHR$(17)
17 CU$=CHR$(145):HO$=CHR$(19)
20 DIMA$(500):Z=65536:GOTO380
23 REM ****************************
**
25 REM *
 *
27 REM *  SIMPLE PROCESADOR DE TEXTO
 *
29 REM *  PARA EL COMMODORRE 64
 *
31 REM *   POR GARY D. MCCLELLAN
 *
33 REM *
 *
35 REM * ADAPTADO DE  'A SIMPLE TEXT
 *
37 REM * PROCESSOR' BY HENRY SIMPSON
 *
39 REM *   MICROCOMPUTING, MAY 1981
 *
41 REM *
 *
43 REM ****************************
**
70 REM **** SUBRUTINA DE MEMORIA DIS
PONIBLE ***
80 M=FRE(0):IFM<0THENM=M+Z
90 PRINT"[5SPC]";CHR$(18)"QUEDAN[SPC]
:";M"[CRSRL][SPC]BYTES":RETURN
95 REM *** RUTINA DE EDICION DE BLOQ
UES DE TEXTO ***
100 PRINTCHR$(147);"---------------
-------------------------";
102 PRINTRO$"REF#";P;RF$:FORX=1TO8:
PRINTCD$;:NEXTX:PRINT
104 PRINT"-------------------------
---------------"
106 PRINT"[2SPC]TECLA[SPC]CLR/HOME:[SPC]
CURSOR[SPC]A[SPC]FIN[SPC]DE[SPC]LINE
A"
108 PRINT"[2SPC]SHIFT[SPC]CLR/HOME:[SPC]
CURSOR[SPC]A[SPC]PRINCIPIO[SPC]DE"
110 PRINT"[SPC]CRSR[SPC]LEFT/RIGHT:[SPC]
MUEVEN[SPC]EL[SPC]CURSOR"
112 PRINT"[2SPC]TECLA[SPC]INST/DEL:[SPC]
BORRA[SPC]CARACTER."
114 PRINT"[2SPC]SHIFT[SPC]INST/DEL:[SPC]
INSERTA[SPC]ESPACIO[SPC]"
116 PRINT"[4SPC]TECLA[SPC]RETURN:[SPC]
VUELVE[SPC]AL[SPC]MODO[SPC]NORMAL"
118 IFA$(P)=""THENA$(P)="[SPC]"
120 PRINTHO$CD$CD$;A$(P)
122 L=1:PRINTHO$CD$CD$RO$;MID$(A$(P)
,L,1);CL$;
124 GETE$:IFE$=""THEN124
126 E=ASC(E$):IFE=13THENGOTO174
128 IFE>31ANDE<96THENGOTO162
130 IFE=29ANDL+1>240THENGOTO124
132 IFE=29THENPRINTRF$;MID$(A$(P),L,
1);:GOTO170
134 IFE=157ANDL-1<1THENGOTO124
136 IFE=157THENPRINTRF$;MID$(A$(P),L
,1);:L=L-1:PRINTCL$CL$;:GOTO172
138 IFE=145ORE=17THENGOTO124
140 IFE=148ANDL+1>240THENGOTO124
142 IFE=147THENPRINTRF$;MID$(A$(P),L
,1);:GOTO122
144 IFE=19THENPRINTRF$HO$CD$CD$;A$(P
);CL$;:L=LEN(A$(P)):GOTO172
146 IFE<>148THENGOTO154
148 PRINTRF$;MID$(A$(P),L,1);CL$;CHR
$(148);"[SPC]";CL$RO$;"[SPC]";CL$;
150 TC$=A$(P):TC$=RIGHT$(TC$,LEN(TC$
)-L+1):A$(P)=LEFT$(A$(P),L-1):E$="[SPC]
"
152 A$(P)=A$(P)+E$+TC$:GOTO124
154 IFE<>20THENGOTO162
156 L=L-1:IFL<1THENGOTO122
158 TC$=A$(P):TC$=RIGHT$(TC$,LEN(TC$
)-L  ):A$(P)=LEFT$(A$(P),L-1)
160 A$(P)=A$(P)+TC$:PRINTCHR$(20);:G
OTO124
162 IFL+1>240THENGOTO124
164 TC$=A$(P):TC$=RIGHT$(TC$,LEN(TC$
)-L):A$(P)=LEFT$(A$(P),L-1)
166 A$(P)=A$(P)+E$+TC$
168 PRINTRF$;MID$(A$(P),L,1);
170 L=L+1:IFMID$(A$(P),L,1)=""THENA$
(P)=A$(P)+"[SPC]"
172 PRINTRO$;MID$(A$(P),L,1);CL$;:GO
TO124
174 PRINTRF$;MID$(A$(P),L,1):RETURN
240 REM **** SUBRUTINA DE ENTRADA DE
 DATOS ***
245 A$(P)=""
250 FORN=1TO240:PRINT"[2SPC][2CRSRL]
[COMM@][CRSRL]";
255 GETB$:IFB$=""THEN255
260 BA=ASC(B$)
265 IFBA=20THENGOSUB1500:PRINTCHR$(1
57);CHR$(32);CHR$(157);:GOTO305
270 IFBA=13THENN=240:GOTO305
280 IFBA=94THENPRINT"[5SPC]";
285 IFBA=43THENPRINT
295 PRINTB$;
300 A$(P)=A$(P)+B$
305 NEXT
310 RETURN
320 REM: **** RUTINA DE VISUALIZACIO
N ***
330 IFA$(P)=""THENRETURN
341 IFASC(LEFT$(A$(P),1))=94THENPRIN
T"[5SPC]";A$(P);:RETURN
351 IFASC(LEFT$(A$(P),1))=43THENPRIN
T:PRINTA$(P);:RETURN
371 PRINTA$(P);:RETURN
380 REM *** MENU PRINCIPAL ***
390 PRINTCHR$(147);CHR$(142):PRINT
405 PRINT"[11SPC]=================="
410 PRINT"[11SPC]=[SPC]MENU[SPC]PRIN
CIPAL[SPC]="
420 PRINT"[11SPC]=================="
425 PRINT
430 PRINT:PRINT"[5SPC]1.[SPC]ESCRIBI
R[SPC]TEXTO"
440 PRINT:PRINT"[5SPC]2.[SPC]REVISAR
/EDITAR[SPC]UN[SPC]BLOQUE"
445 PRINT:PRINT"[5SPC]3.[SPC]INSERTA
R/COPIAR[SPC]UN[SPC]BLOQUE"
450 PRINT:PRINT"[5SPC]4.[SPC]LEER[SPC]
DEL[SPC]DISCO"
460 PRINT:PRINT"[5SPC]5.[SPC]GUARDAR
[SPC]EN[SPC]DISCO"
470 PRINT:PRINT"[5SPC]6.[SPC]SACAR[SPC]
POR[SPC]IMPRESORA"
480 PRINT:PRINT"[5SPC]7.[SPC]TERMINA
R"
484 PRINT:GOSUB70
485 PRINT:PRINT"[5SPC]SELECCIONAR[SPC]
(1-7):"
490 GETAO$:IFAO$=""THEN490
500 IFAO$="1"THEN570
510 IFAO$="2"THEN680
515 IFAO$="3"THENGOSUB2100
520 IFAO$="4"THENGOTO 1140
530 IFAO$="5"THENGOSUB1200
540 IFAO$="6"THENGOSUB1060
550 IFAO$="7"THENCLOSE200:END
560 GOTO380
```

```
570 REM **** RUTINA DE ESCRITURA ***
*
580 FORG=1TO500
590 IFA$(G)=""THENF=G:G=500
600 NEXT
602 PRINTCHR$(147)
603 PRINT"[4SPC]ENTRA[SPC]EL[SPC]NUM
ERO[SPC]DE[SPC]REFERENCIA[SPC]O"
604 PRINT"[SPC]´RETURN´[2SPC]PARA[SPC]
LA[SPC]PROXIMA[SPC]DISPONIBLE":INPUT
F
605 IFF<1ORF>500THENPRINT"DE[SPC]1[SPC]
A[SPC]500[SPC]POR[SPC]FAVOR":GOTO603

610 PRINTCHR$(147)"------------------
------------------------";CHR$(14);
620 FORP=FTO500
630 PRINTCHR$(18);P;CHR$(146);
635 IFA$(P)<>""THENPRINTCHR$(18);"RE
F.[SPC]OCUPADA":FORX=1TO100:NEXT:GOT
O660
640 GOSUB240
650 IF(A$(P))=""THENP=500
660 NEXT
670 GOTO380
680 REM *** REVISA/CORRIGE PANTALLA
***
690 PRINTCHR$(147)
700 INPUT"ENTRA[SPC]REF#[SPC]A[SPC]R
EVISAR:";P
710 IFP<1ORP>500THENPRINTPO$"DE[SPC]
1[SPC]A[SPC]500[SPC]POR[SPC]FAVOR-"R
F$:GOTO700
720 REM **** VISUALIZA TEXTO REF ****
725 PRINTCHR$(147);"----------------
-----------------------";
730 PRINTCHR$(14);RO$;"REF#";P;RF$
740 IFA$(P)<>""THEN800
760 PRINTCHR$(18);"EMPTY"
780 GOTO820
800 GOSUB320
820 PRINTHO$:FORX=1TO8:PRINTCD$;:NEX
T
830 PRINT"-------------------------
--------------";
835 PRINTRO$"TECLAS[SPC]DE[SPC]CONTR
OL";RF$
840 PRINT"[SPC]´F´:[SPC]ADELANTE"
850 PRINT"[SPC]´B´:[SPC]ATRAS"
860 PRINT"[SPC]´X´:[SPC]SALIR";
865 PRINTSPC(8)RO$"ESCOGE[SPC]?";RF$

870 PRINT"[SPC]´D´:[SPC]BORRAR"
880 PRINT"[SPC]´E´:[SPC]EDITAR"
900 PRINTRO$"CODIGOS[SPC]DE[SPC]CONT
ROL:"RF$
905 PRINT"[10SPC]´@´[SPC]:[SPC]DOBLE
[SPC]ANCHO"
910 PRINT"[10SPC]´↑´[SPC]:[SPC]NUEVO
[SPC]PARRAFO"
920 PRINT"[10SPC]´+´[SPC]:[SPC]SALTA
R[SPC]LINEA"
930 PRINT"[10SPC]´←´[SPC]:[SPC]MARGE
N[SPC]IZQUIERDO"
941 GETAO$:IFAO$=""THEN941
945 PRINTCHR$(147)
950 IFAO$="X"THEN380
960 IFAO$="F"ANDP<500THENP=P+1:GOTO720
```

```
970 IFAO$="F"THEN720
980 IFAO$="B"ANDP>1THENP=P-1:GOTO720

990 IFAO$="B"THEN720
1000 IFAO$="E"THENGOSUB100:GOTO720
1010 IFAO$="D"THENA$(P)="":GOTO720
1020 GOTO725
1060 REM **** HACE COPIA IMPRESA ***
*
1061 PRINTCHR$(147)
1062 PRINT"RUTINA[SPC]DE[SPC]CONFIGU
RACION[SPC]":INPUTQ$:IFLEFT$(Q$,1)<>
"Y"THEN1085
1063 PRINT"ENTRA[SPC]MARGEN[SPC]IZQU
IERDO[SPC](NUMERO[SPC]DE[SPC]CARACTE
RES)"
1064 LM=5:INPUTLM:IFLM>70THENPRINT"1
[SPC]A[SPC]70[SPC]POR[SPC]FAVOR":GOT
O1063
1065 PRINT"ENTRA[SPC]LONGITUD[SPC]DE
[SPC]LINEA[SPC](NUMERO[SPC]TOTAL[SPC]
DE[SPC]CARACTERES)":LL=70:INPUTLL
1066 IFLL<20ORLL>80THENPRINT"DE[SPC]
20[SPC]A[SPC]80[SPC]POR[SPC]FAVOR-":
GOTO1065
1067 PS=3:PRINT"ENTRA[SPC]MARGEN[SPC]
SUPERIOR[SPC](NUMERO[SPC]DE[SPC]LINE
AS)"
1069 INPUT PS:IFPS<1ORPS>65THENPRINT
"DE[SPC]1[SPC]A[SPC]65[SPC]POR[SPC]F
AVOR":GOTO1067
1071 PE=3:PRINT"ENTRA[SPC]MARGEN[SPC]
INFERIOR[SPC](NUMERO[SPC]DE[SPC]LINE
AS)"
1073 INPUTPE:IFPE<1ORPE>65THENPRINT"
DE[SPC]1[SPC]A[SPC]65[SPC]POR[SPC]FA
VOR":GOTO1071
1075 RP=1:PRINT"ENTRA[SPC]REF#[SPC]I
NICIAL"
1077 INPUTRP:IFRP<1ORRP>500THENPRINT
"DE[SPC]1[SPC]A[SPC]500[SPC]POR[SPC]
FAVOR":GOTO1075
1083 RE=500:PRINT"ENTRA[SPC]REF#[SPC]
FINAL":INPUTRE:IFRE<1ORRE>500THENGOT
O1075
1085 PRINT"PREPARA[SPC]EL[SPC]PAPEL[SPC]
DE[SPC]LA[SPC]IMPRESORA.":PRINT"PULS
A[SPC]UNA[SPC]TECLA[SPC]PARA[SPC]SEG
UIR"
1087 GETQ$:IFQ$=""THEN1087
1089 REM **** ACTIVA IMPRESORA ****
1091 OPEN4,4,7:CMD4:GOSUB1387
1093 LM$="":FORJ=1TOLM:LM$=LM$+CHR$(
32):NEXTJ:PF=0
1095 FORP=RPTORE
1097 IFA$(P)=""ANDA$(P+1)=""THENP=50
0:GOTO1110
1100 GOSUB1300
1110 NEXT
1120 IFPF=1THENPO$=LM$+B$:PRINTPO$
1125 PRINT#4:CLOSE4
1130 RETURN
1140 REM **** LEE ARCHIVO DEL DISCO
***
1142 PRINTCHR$(147):PRINT"CARGANDO[SPC]
EL[SPC]ARCHIVO[SPC]DE[SPC]TEXTO";
1145 OPEN15,8,15
1150 OPEN5,8,5,"0:TEXTFILE,S,R"
1152 GOSUB2000:IFER=1THENER=0:FI=0:G
OTO1180
1155 INPUT#5,FI
1160 FORP=1TOFI:A$(P)=""
1162 GET#5,C$:IFASC(C$)<>13THENA$(P)
=A$(P)+C$:GOTO1162
1175 PRINT".";:NEXT
1180 CLOSE5:CLOSE15
1182 PRINT:PRINT:PRINTFI;"BLOQUES[SPC]
DE[SPC]TEXTO[SPC]LEIDOS.":PRINT"PULS
A[SPC]UNA[SPC]TECLA"
1184 GETQ$:IFQ$=""THEN1184
1190 GOTO380
1200 REM **** GUARDA ARCHIVO DE TEXT
O ****
1202 PRINTCHR$(147):PRINT"GUARDANDO[SPC]
ARCHIVO[SPC]DE[SPC]TEXTO":PRINT
1205 OPEN15,8,15
1210 OPEN5,8,5,"@0:TEXTFILE,S,W"
1215 GOSUB2000:IFER=1THENER=0:FI=0:G
OTO1275
1240 FORP=1TO500
1245 IFA$(P)<>""THENFI=P
1250 NEXTP
1255 PRINT#5,FI
1260 FORP=1TOFI
1265 PRINT#5,A$(P)
1270 NEXTP
1275 CLOSE5:CLOSE15
1280 PRINTFI;"BLOQUES[SPC]DE[SPC]TEX
TO[SPC]GUARDADOS.":PRINT"PULSA[SPC]U
NA[SPC]TECLA"
1285 GETQ$:IFQ$=""THEN1285
1290 RETURN
1300 REM **** RUTINA DE IMPRESORA **
**
1305 IFA$(P)=""THENRETURN
1310 TC=ASC(LEFT$(A$(P),1))
1315 IFTC=94THENGOSUB1380:CC=LM+5:B$
="[5SPC]":PF=1:GOTO1340
1320 IFTC=43THENGOSUB1380:PRINT:LC=L
C+1:GOSUB1383:GOTO1340
1325 IFTC=64THENGOSUB1380:DW=1:PRINT
CHR$(14):GOSUB1383:GOTO1340
1330 IFTC=95THENGOSUB1380:GOTO1340
1335 TC$=A$(P):GOTO1345
1340 TC$=RIGHT$(A$(P),LEN(A$(P))-1):
IFTC$=""THENPF=0:RETURN
1345 IFPF=0THENB$=""
1346 FORX=1TOLEN(TC$):IFB$=""ANDMID$
(TC$,X,1)=CHR$(32)THENCC=CC+1:X=X+1
1350 CC=CC+1:B$=B$+MID$(TC$,X,1)
1355 IFLL-CC<10ANDMID$(TC$,X,1)="[SPC]
"THENPF=1:GOSUB1380
1360 IFCC>LLTHENPF=1:GOSUB1380
1365 NEXTX:IFB$<>""THENPF=1
1375 RETURN
1380 IFPF=1ANDB$<>""THENPO$=LM$+B$:P
RINTPO$:B$="":CC=LM:LC=LC+1
1381 IFDW=1THENPRINTCHR$(15);:DW=1
1383 IFLC<(66-PE)THEN1399
1385 FORI=1TOPE:PRINT:NEXTI
1387 FORI=1TOPS:PRINT:NEXTI
1389 LC=PS
1399 PF=0:RETURN
1500 REM **** RUTINA DE TECLA DELETE
 PARA BLOQUE ESCRITO ****
1505 IFLEN(A$(P))<1THENA$(P)="":B$="
":N=240:GOTO1530
1510 A$(P)=LEFT$(A$(P),LEN(A$(P))-1)

1520 IFN>1ORN=1THENN=N-1
1530 RETURN
2000 REM **** RUTINA DE ERRORES DE D
ISCO ***
2010 INPUT#15,S1$,S2$,S3$,S4$
2020 IFVAL(S1$)>0THENPRINTS1$,S2$,S3
$,S4$:CLOSE5:CLOSE15:ER=1
2030 RETURN
2100 PRINTCHR$(147)
2110 PRINT"'I'[SPC]INSERTA[SPC]UN[SPC]
BLOQUE[SPC]VACIO[SPC]EN[SPC]REF#"
2120 PRINT"'C'[SPC]COPIA[SPC]UN[SPC]
BLOQUE[SPC]EN[SPC]OTRO[SPC]NUEVO[SPC]
REF#"
2130 INPUT"COMANDO:";Q$
2140 IFQ$="C"THEN2200
2150 IFQ$="I"THEN2400
2160 GOTO380
2200 REM COPIAR BLOQUE
```

```
2210 PRINT:PRINT"COPIAR[SPC]DEL[SPC]
BLOQUE[SPC]REF#:"
2220 INPUTP1:IFP1<00RP1>500THEN2210
2225 INPUT"AL[SPC]BLOQUE[SPC]REF#:";
P
2230 IFP<00RP>500THEN2210
2240 PRINT"REF#";P1;"[SPC]SERA[SPC]C
OPIADO[SPC]EN[SPC]REF#";P:PRINT"ESTA
S[SPC]SEGURO[SPC]?"
2330 Q$="N":INPUTQ$:IFLEFT$(Q$,1)="S
"THENGOTO2350
2340 GOTO380
2350 A$(P)=A$(P1)
2360 PRINT"OTRA[SPC]COPIA[SPC]?:"
2370 Q$="N":INPUTQ$:IFLEFT$(Q$,1)="S
"THENGOTO2210
2380 GOTO380
2400 REM INSERTAR TEXTO EN BLANCO EN
 EL BLOQUE REF#
2410 PRINT:PRINT"REF#[SPC]DONDE[SPC]
INSERTAR[SPC]EL[SPC]BLOQUE[SPC]?:"
2420 INPUTP1:IFP1<1ORP1>500THENGOTO2
410
2430 FORP=P1TO500:IFA$(P)=""ANDA$(P-
1)=""THENP2=P-1:P=500
2440 NEXTP
2445 IFP2<1THENP2=1
2450 FORX=P2TOP1STEP-1:A$(X)=A$(X-1)
:NEXTX
2455 A$(P1)=""
2460 PRINT"REF#[SPC]";P1;"[SPC]LIBRE
[SPC]PARA[SPC]ENTRAR[SPC]TEXTO"
2470 Q$="N":INPUT"INSERTO[SPC]OTRO[SPC]
BLOQUE";Q$
2480 IFLEFT$(Q$,1)="S"THEN2410
2490 GOTO380
5000 REM *** LISTA DE VARIABLES DEL
PROGRAMA ****
5010 REM RP=REF INICIAL    RE= REF F
INAL     LL= LONGITUD DE LINEA
5020 REM PS=COMIENZO DE PAGINA    PE
= FIN DE PAGINA
5025 REM LM$=STRING DE MARGEN IZQUIE
RDO   HO$=TECLA HOME    CL$=CURSOR IZ
QUIERDA
5030 REM CR$=CURSOR DERECHA   CD$=CU
RSOR ABAJO   CU$= CURSOR ARRIBA
5040 REM Q$=VARIABLE TEMPORAL PARA P
REGUNTAS
5050 REM A$(P)=BLOQUE DE LA MATRIZ D
E TEXTO   P=PUNTERO DE LA MATRIZ
5055 REM P1= PUNTERO TEMPORAL
5060 REM FI=VARIABLE TEMPORAL PARA M
ANEJO DE FICHEROS
5065 REM AO$=VARIABLE DE CADENA PARA
 COMANDOS DEL MENU
5070 REM S1$ A S4$=VARIABLES PARA MA
NEJO DE ERRORES
5075 REM B$=STRING DE TRABAJO    TC$
=STRING TEMPORAL DE TRABAJO
5080 REM RO$=REVERSE ON        RF$=REV
ERSE OFF
5090 REM E$=STRING DE TRABAJO EN EDI
CION   C$=IDEM PARA MANEJO DE ARCHIV
OS
5100 REM PO$=IDEM PARA IMPRIMIR
5105 REM PF=INDICADOR (FLAG) DE ESTA
DO DE CADENAS DE TRABAJO
```

```
5110 REM N=CONTADOR    E=VALOR ASCII
PARA EL MODO DE EDICION
5120 REM L=PUNTERO PARA LA FUNCION M
ID$    G=CONTADOR PARA LAZOS
5130 REM DW=INDICADOR (FLAG) DE DOBL
E ANCHO    ER=INDICADOR DE ERROR
5135 REM F=CONTADOR
```

SUMA DE CONTROL DEL PROCESADOR DE TEXTO

2	0	10	71	15	230
17	73	20	126	23	197
25	163	27	156	29	121
31	132	33	163	35	33
37	212	39	249	41	163
43	197	70	174	80	36
90	109	95	195	100	15
102	209	104	229	106	167
108	3	110	8	112	75
114	177	116	220	118	74
120	235	122	50	124	108
126	99	128	196	130	209
132	176	134	161	136	184
138	242	140	3	142	222
144	18	146	155	148	126
150	218	152	144	154	95
156	224	158	134	160	5
162	192	164	70	166	54
168	193	170	73	172	18
174	78	240	222	245	252
250	249	255	107	260	178
265	226	270	188	280	140
285	103	295	58	300	206
305	130	310	142	320	204
330	188	341	165	351	83
371	162	380	226	390	10
405	135	410	238	420	135
425	153	430	86	440	127
445	209	450	129	460	30
470	33	480	55	484	199
485	184	490	8	500	169
510	172	515	95	520	127
530	97	540	102	550	255
560	36	570	141	580	228
590	44	600	130	602	77
603	138	604	138	605	112
610	140	620	2	630	206
635	231	640	35	650	22
660	130	670	36	680	92
690	77	700	197	710	46
720	168	725	15	730	23
740	120	760	40	780	35
800	34	820	155	830	32
835	63	840	41	850	98
860	179	865	181	870	177
880	163	900	138	905	20
910	251	920	96	930	218
941	9	945	77	950	207
960	242	970	187	980	137
990	183	1000	155	1010	120
1020	39	1060	13	1061	77
1062	186	1063	250	1064	39
1065	226	1066	124	1067	88
1069	141	1071	47	1073	62
1075	79	1077	189	1083	57
1085	221	1087	189	1089	201
1091	59	1093	29	1095	96
1097	209	1100	81	1110	130
1120	131	1125	218	1130	142
1140	158	1142	24	1145	251
1150	213	1152	230	1155	116
1160	29	1162	154	1175	2
1180	21	1182	252	1184	187
1190	36	1200	124	1202	85
1205	251	1210	26	1215	235
1240	237	1245	113	1250	210
1255	136	1260	231	1265	255
1270	210	1275	21	1280	48
1285	189	1290	142	1300	141
1305	188	1310	220	1315	159
1320	20	1325	103	1330	205
1335	3	1340	12	1345	6
1346	156	1350	178	1355	151
1360	13	1365	205	1375	142
1380	198	1381	186	1383	71
1385	190	1387	204	1389	228
1399	64	1500	124	1505	30
1510	249	1520	109	1530	142
2000	14	2010	64	2020	151
2030	142	2100	77	2110	186
2120	137	2130	180	2140	164
2150	172	2160	36	2200	85
2210	165	2220	18	2225	35
2230	112	2240	203	2330	224
2340	36	2350	239	2360	56
2370	219	2380	36	2400	185
2410	244	2420	158	2430	93
2440	210	2445	253	2450	175
2455	45	2460	170	2470	220
2480	45	2490	36	5000	98
5010	108	5020	52	5025	229
5030	90	5040	72	5050	203
5055	62	5060	127	5065	77
5070	98	5075	148	5080	90
5090	47	5100	155	5105	167
5110	139	5120	209	5130	107
5135	140				



Ayuda para teclear los programas



RUN DANSK (Dinamarca)

Traducido por Diego ROMERO y adaptado por Alvaro IBAÑEZ

Dado que muchas de las llamadas que recibimos sobre pegas al poner en marcha los programas de la revista son causadas por errores al teclear los programas frente a una minoría causada por errores nuestros (todo hay que reconocerlo), creo que es interesante este corto programa que ayudará a comprobar que lo habéis tecleado exactamente igual que el original.

Este programa va a ser tan estricto con la forma de teclear el programa que no permitirá añadir ni quitar ni un solo espacio. La idea y el programa original lo hemos sacado de la hermana de Commodore World en Dinamarca (RUN DANSK), y como no podemos traducirlo literalmente porque nuestro Danés no es perfecto, voy a hacer una pequeña traducción libre.

Todos los programas que aparezcan en la revista Commodore World (o la mayoría si el espacio no lo permite), irán acompañados de una "Suma de Control", que obligará a ceñirse estrictamente al listado original.

Debéis seguir una serie de pasos que permitirán localizar las líneas en que cometáis errores. Lo primero que debéis hacer es teclear el programa de control de suma y guardarlo en cinta o disco para utilizarlo siempre. Los pasos a seguir para introducir un programa son:

1.—CARGAR el programa "control de suma", que previamente habíais grabado.

2.—TECLEAR el programa que queréis copiar.

3.—SALVAR el programa antes de probarlo.

4.—TECLEAR GOTO 62000 y contestar a las preguntas que hace el programa de control de suma.

5.—COMPARAR las sumas de cada línea de vuestro programa con el que aparece en la revista. La línea o líneas que no coincidan deberéis repasarlas, ya que no serán idénticas al original.

6.—SALVAR en cinta o disco una copia del programa corregido antes de ejecutarlo.

7.—UTILIZAR el programa que ya funcionará correctamente con sólo hacer RUN.

NOTA: Posiblemente en este número no exista tiempo material para preparar las tablas de sumas de control de los programas publicados antes de llevarlo a imprenta (dentro de media hora), pero en el próximo será de gran ayuda, os lo puedo asegurar.

VIC-20 ó C-64
con o sin ampliación

```
62000 REM SUMA DE CONTROL PARA VIC-2
0 & C-64
62010 PRINT"[CLR]PARA[SPC]QUE[SPC]OR
DENADOR?"
62020 INPUT"VIC20=0[SPC]CBM64=1";CO
62030 GOSUB 62070
62040 GOTO 62240
62050 IF FL>=0 THEN 62040
62060 END
62070 DEFFN DEEK(X) = PEEK(X)+256*PE
EK(X+1)
62080 DATA ***
62090 DATA 165,252,166,253,133,020,1
34,021,032,019
62100 DATA 166,216,160,001,177,095,1
33,254,240,013
62110 DATA 200,177,095,133,252,200,1
77,095,133,253
62120 DATA 200,169,000,133,251,177,0
95,240,006,024
62130 DATA 101,251,200,208,244,096
62140 DATA -1
62150 AD = 52992:IFCO=0THENAD=828
62160 RESTORE
62170 READ T$: IF T$<>"***" THEN 621
70
62180 READ T : IF T>=0 THEN POKE AD,
T : AD = AD+1 : GOTO 62180
62190 IFCO=0THENPOKE838,198
62200 PRINT"[CLR]DONDE[SPC]QUIERE[SPC]
QUE[SPC]LO[3SPC]IMPRIMA:"
62210 PRINT"TV/MONITOR=0[SPC]PLOTTER
=6[SPC]IMPRESORA=4"
62220 INPUT DEV
62230 RETURN
62240 FL = 0 : INPUT "LINEA[SPC]INIC
IAL"; FL : IF FL<0 THEN RETURN
62250 LL = 65536 : INPUT "LINEA[SPC]
FINAL", LL:PRINT"[CLR]"
62260 IF DEV>0 THEN OPEN 1,DEV
```

```
62270 LN = FL : C = 0 : C1 = 0
62280 POKE 252,LN-INT(LN/256)*256 :
POKE 253,LN/256
62290 IFCO=0THEN SYS828 :CS=PEEK(251
) LN=FNDEEK(252)+1
62300 IFCO=1THEN SYS 52992:CS=PEEK(2
51) LN=FNDEEK(252)+1
62310 T$ = LEFT$(STR$(LN-1)+"[6SPC]"
,6)+LEFT$(STR$(CS)+"[7SPC]",5)
62320 IF DEV=0 THEN PRINT T$;
62330 IF DEV>0 THEN PRINT#1,T$;
62340 IF DEV=0 THEN C = C+1 : IF C>=
3 THEN PRINT : C = 0 : C1 = C1 + 1
62350 IF DEV>0 THEN C = C+1 : IF C>=
3 THEN PRINT#1 : C = 0 : C1 = C1 + 1

62360 IF LN<=LL AND PEEK(254) THEN 6
2280
```

```
62370 IF DEV>0 THEN PRINT#1:CLOSE1
62380 END
```

```
SUMA DE CONTROL DEL PROGRAMA DE CONTROL
 62000 228  62010 156  62020 17
 62030 172  62040 167  62050 179
 62060 128  62070 179  62080 33
 62090 30   62100 38   62110 46
 62120 31   62130 14   62140 1
 62150 59   62160 140  62170 169
 62180 31   62190 174  62200 52
 62210 75   62220 132  62230 142
 62240 149  62250 134  62260 77
 62270 13   62280 207  62290 92
 62300 198  62310 187  62320 191
 62330 250  62340 114  62350 161
 62360 200  62370 38   62380 128
 62381 0
```

COMMENTARIOS COMMODORE WORLD

BASIC EN CASTELLANO PARA EL COMMODORE 64

La casa Sakati, S. A. de Madrid ha lanzado este mes el programa que permite trabajar en castellano con el Commodore 64. A nosotros nos ha enviado la versión en disco, pero también lo comercializan en cinta.

Este programa se une a la nueva y creciente ola de intentos de españolizar los ordenadores, a la que estamos asistiendo, baste recordar que el mes



pasado sacamos reseña del LOGO en castellano que presentaba otra casa de Madrid y Barcelona. Parece ser que la tendencia actual es la de cambiar los lenguajes tradicionales de programación y particularizar las versiones para España.

Las instrucciones que acompañan al programa son cortas, pues no pretenden ser un manual de programación en Basic, sino una sencilla explicación de cómo cargar el programa (utilizando los comandos LOAD y RUN o las teclas Shift y Stop) y el modo de cambiar entre el español y el Basic original, que se limita a un simple comando POKE. El resto de la hoja de instrucciones son dos tablas que indican los comandos originales y su equivalente actual, así como los antiguos y nuevos mensajes de error.

Para citar un ejemplo de la traducción de los comandos podemos decir que NEW se sustituye por INI, GOTO por VEAL, STEP por PASO, RIGHT$ por DRCH$, INPUT por INTRO, etc.

En la página 31 del número cuatro de Commodore World hemos publicado una colaboración de Guillermo Caamaño Olivares de Granada que hacía algo similar, pero en ella debíamos escoger los nuevos comandos uno a uno nosotros mismos, cosa que se hace un poco pesada, y aquellos que prefieran tenerlo todo resuelto tienen la oportunidad de adquirir el programa que ha lanzado Sakati.

Un pequeño detalle que está omitido en las instrucciones del programa y que ya advertía Guillermo en su colaboración, es que al cambiar el Basic al castellano quedan prohibidas automáticamente todas las variables que coincidan con los comandos del nuevo basic, por ejemplo: no podemos utilizar "SI", ya que es el equivalente del antiguo "IF".

Aquellos a los que les parezca interesante el programa y deseen adquirirlo pueden dirigirse a:

Sakati, S. A.
Ardemans, 24
28028 Madrid
Tel. (91) 256 77 94

Otra novedad de esta casa de la que esperamos poder hacer reseña próximamente es un magnífico programa para la resolución de quinielas que permite imprimir los resultados directamente en el formato de los boletos del patronato o en papel continuo o pantalla, pero aún no han salido de imprenta las instrucciones para poder entender este programa, y al ser tan completo y lleno de posibilidades he de confesar que me he perdido intentando adivinar cómo usarlo para adelantarme a los demás y ser el primero en conseguir "una de catorce".

Un secreto: espero que no se entere nadie y yo sea el único que acierte el pleno una de las próximas semanas, por eso prefiero callar la fecha de lanzamiento de este programa y poder aprovecharme de él yo solito sin compartirlo con nadie más.

SOBRE MONITORES

En el SIMO hemos tenido oportunidad de probar los monitores de la marca Tatung UK (Reino Unido), que distribuye en España la casa Alpha Mundial Group, S. A. de Barcelona.

El monitor de color en alta resolución es el que más destaca. En primer lugar por su definición, que es de 584 puntos verticales por 436 horizontales en su pantalla de 14", siendo la superficie útil para visualización de 275*201 mm. y estando recomendado para gráficos de alta resolución y/o texto en formato de 25 líneas de 40 caracteres (ideal para el Commodore). Dispone de entradas YUV, RGB lineal y TTL, además de la entrada PAL, pudiéndose utilizar, por tanto, con una amplia gama de ordenadores (Apple, Sinclair, Amstrad, y por supuesto los Commodore).

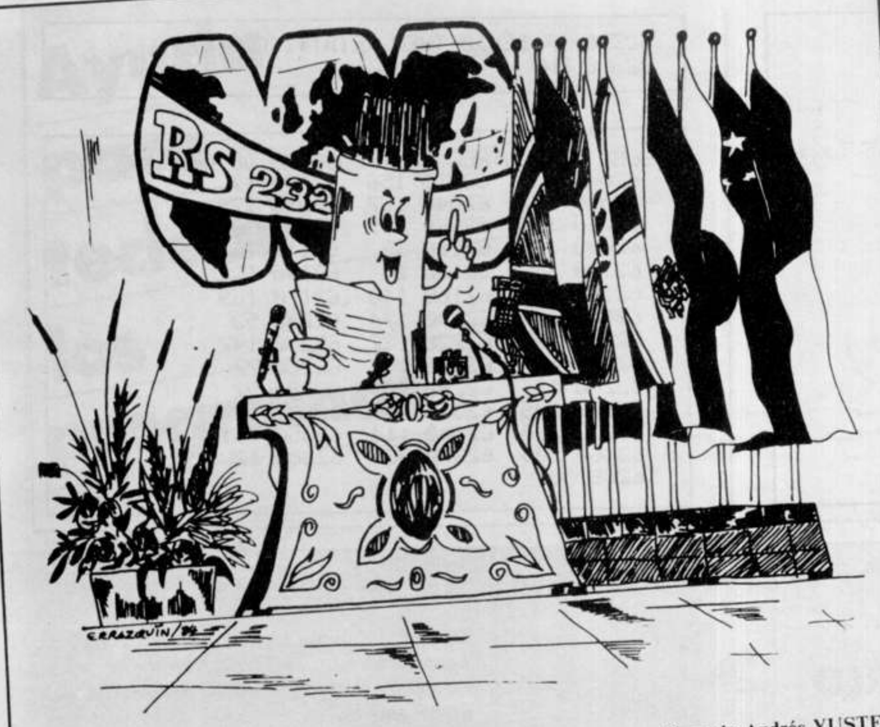
La estabilidad y nitidez de la imagen en pantalla era perfecta (muy superior a otros monitores), y además dispone de un conmutador en la parte trasera que permite hacerlo funcionar como monitor verde con mayor resolución, con lo que la vista descansa más cuando deben pasarse muchas horas delante de la pantalla tecleando programas "serios" sin colores.

Su aspecto físico es bueno, no siendo voluminoso, y por fin lo más interesante... su precio es de sólo 59.500 pesetas. Creo que es uno de los más económicos del mercado nacional dentro de sus buenas prestaciones.

El otro monitor era monocromático con pantalla de 12" verde antirreflexiva de fósforo P31, su resolución era superior al de color, ya que son 900 líneas verticales y está recomendado para utilizar hasta con 2000 caracteres en pantalla, siendo ideal para completar las tarjetas de 80 columnas de todos aquellos que quieran destinar su equipo a aplicaciones más serias. El precio de este monitor es de 25.000 pesetas.

Ambos monitores de la marca Tatung son comercializados por la compañía:

Alpha Mundial Group, S. A.
Gran Vía Carlos III, 86, 6
08028 Barcelona
Tel. (93) 330 96 51



Jim GRUBBS (RUN EEUU)

Traducido y adaptado por Eugenio Andrés YUSTE

Pasaporte al mundo (II)

Amplía tus horizontes sobre computadores con este artículo que os abrirá nuevas puertas a tí y a tu Commodore. La clave está en la puerta del usuario de tu Commodore.

Si dispones de una salida RS-232 vacante en tu ordenador Commodore 64 ó VIC-20, lee este artículo para aprender cómo se puede utilizar esta conexión de forma que incremente la eficacia de tu ordenador.

Los equipos periféricos, al igual que cualquier ampliación del sistema han de conectarse a tu VIC-20 ó C-64 de manera que se puedan comunicar con la tarjeta principal de proceso del ordenador. Los fabricantes se ven libres para diseñar cualquier tipo de interconexión para comunicar con la tarjeta principal, la cual normalmente se basa en un standard ya establecido.

Existen varias puertas de entrada al ordenador. Incluso la salida que se utiliza para conectar el Joystick al ordenador es un interface que nos permite mandar información de posicionamiento al ordenador.

La salida serial (llamada también bus serial), es otra de las formas corrientes para conectar periféricos al ordenador. Por lo general se utiliza para conectar unidades de discos o impresoras, pero también se podría usar para conectar otros aparatos.

El ordenador también ofrece otro tipo de salida, la salida de ampliación (también llamada bus paralelo). A esta salida se conecta el cartucho de memoria adicional en el VIC-20, y donde se conectan los cartuchos de juegos. Existen varias cosas que podríamos investigar en torno a este tipo de salida, pero en este artículo investigaremos "la salida del usuario", es decir, la salida RS-232.

Si miramos al ordenador por detrás, este conector (RS-232) está situado a la extrema derecha. A continuación veremos las funciones que podemos realizar con esta conexión.

La importancia del RS-232

A lo mejor ya estás utilizando esta conexión para acoplar un VIC-Modem a tu ordenador o una impresora del tipo RS-232.

La ampliación más común del VIC-20 o el C-64 es el modem, que tiene que ir conectado a la salida RS-232 para poder usarlo. Pero; ¿qué es un modem y cómo es el software que lo acompaña?

Un modem no es más que un aparato que actúa de interface entre el ordenador y un teléfono. De esta manera un ordenador se puede comunicar con otro ordenador. El ordenador utiliza un lenguaje digital que consiste en rápidos cambios de voltaje mientras que el teléfono trabaja con sonidos que están comprendidos en la gama de la frecuencia de la voz. De ahí, que el modem se encargue de convertir las señales digitales del ordenador a señales acústicas de manera que se puedan enviar a través de las líneas telefónicas.

Así que ¡ya estamos preparados una vez que hayamos conectado el modem! Pues... no del todo. Tu ordenador trabaja a una velocidad demasiado rápida para las líneas telefónicas, por lo cual tendrás que incorporar un programa controlador que se encargue de formatear los datos de manera que puedan pasar eficazmente a la línea telefónica mediante el modem.

Además, tu Commodore habla a través de una versión ampliada del ASCII, y por lo tanto si quieres comunicarte con máquinas que utilicen el ASCII standard, tendrás que hacer unas pequeñas modificaciones. Asimismo quizás quieras almacenar la información que recibas a través de tu modem.

Estas tareas las puedes realizar mediante un programa de terminal. Un programa de terminal es aquel que formatea datos y controla la velocidad mas otra serie de factores a través del software. Manda una señal digital al modem para convertirlas en tonos cuando se manden datos, y hace lo opuesto cuando estés recibiendo datos.

Estableciendo los standards

¿Cómo apareció el término RS-232? El término apareció cuando un grupo de prestigiosos ingenieros establecieron los standards para las conexiones necesarias que permitieran a los ordenadores no tan solamente comunicarse entre sí, sino que también pudieran comunicarse con otros

aparatos tales como las impresoras, unidades de disco y otros periféricos. Desafortunadamente el RS-232 no es el único standard. Existen otros standards establecidos tal como el IEEE-488. El VIC-20 y el C-64 no pueden llegar a estos standards sin algún complemento de ayuda.

En realidad, las máquinas de Commodore no son capaces de realizar una verdadera comunicación RS-232 ya que el standard Commodore RS-232 es ligeramente distinto al verdadero standard RS-232, pero las limitaciones se solventan fácilmente. Ocurre lo mismo que con las diversas variaciones del viejo standard "Kansas City" para el almacenamiento de datos sobre cassettes.

Esto se debe a que gran parte de la tecnología de circuitos integrados utilizados actualmente en los ordenadores emplean una lógica de Transistor-Transistor (TTL). Los aparatos TTL representan los dos estados binarios, dando una señal positiva de 5 V para representar el 1 y una señal de tierra, OV, para representar el número binario 0.

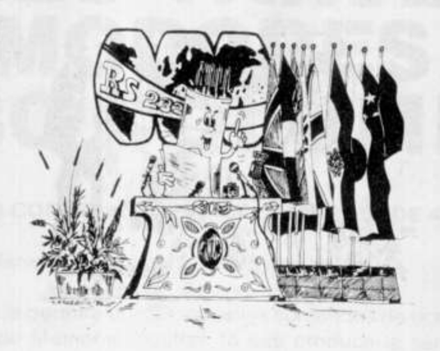
La casa Commodore decidió utilizar las señales TTL en sus ordenadores para las salidas RS-232. Los verdaderos standards del RS-232 son de señales positivas entre 9 y 12 V para el 1 y una señal negativa entre 9 y 12 V para el 0. En el mercado existen varios interfaces que se encargan de solventar este pequeño problema de diferencia de voltajes en caso de que se deseara conectar algún aparato que se rija por el standard de +9-12 V y —9-12 V.

Si utilizas un producto fabricado por la casa Commodore, tal como el VIC-Modem, entonces no tienes que molestarte en ajustar el voltaje. Sin embargo, si se desea utilizar un producto que no haya sido fabricado por Commodore, entonces lo más probable es que necesitara un interface que se encargue de neutralizar la diferencia de voltaje.

Físicamente, las conexiones a la salida RS-232, se hacen con un conector de 24 patillas. Estas pueden ser algo difíciles de conseguir. (En la tabla 3 hay una dirección en donde se podría conseguir un conector de este tipo). Si no eres capaz de encontrar alguna tienda que los venda, siempre te quedará la solución de poder fabricarlo tu mismo. Eso sí, tendrás que tomar las debidas precauciones cuando lo instales asegurándote que las patillas estén alineadas adecuadamente. ¡Siempre que realices esta operación asegúrate que el ordenador esté desenchufado!

Función OPEN

La clave de la utilización de las funciones RS-232 está en la sentencia OPEN del programa controlador de comunicaciones que la casa Commodore ha facilitado con sus ordenadores. Esta sentencia es parecida a la de abrir un fichero de disco o cassette, sólo que en este caso la parte del fichero que se abre es la oficina electrónica del ordenador, que viene etiquetada RS-232.



Existen varios factores importantes a tener en cuenta para abrir un canal RS-232. En primer lugar tenemos que considerar que sólo disponemos de uno. Si intentásemos abrir otro, lo único que conseguiremos es recolocar los "pointers" (punteros) obteniendo nuevamente un solo canal. Por lo tanto, lo primero que habremos de hacer en nuestro programa es abrir un canal RS-232.

Al abrir un canal RS-232 ocurren dos cosas. En primer lugar, se ejecuta la instrucción Clear, lo que significa que todas las variables así como la memoria se posicionan en cero. En segundo lugar el ordenador reserva 512 bytes de memoria para los "buffers" de recepción y transmisión. No causa ningún efecto si sólo se desea usar la salida RS-232 para comunicaciones en un solo sentido, los 512 bytes se reservan igualmente. En caso de que no se disponga de suficiente espacio en memoria, no existe ninguna indicación de alerta por parte del programa, por lo cual es conveniente llevar la cuenta de los bytes utilizados.

Con unos cuantos comandos básicos más, ya estarás preparado para escribir el software de terminal que necesitarás para comunicarte a través de la salida RS-232

La sentencia OPEN es la siguiente:

OPEN lfn,2,0, (Registro de control) (registro de comando) (Baudios bajos opcional) (Baudios altos opcional). A continuación explicaré lo que significa esta sentencia.

Al igual que con todos los ficheros, se ha de elegir un número de fichero entre el 1 y el 255 (lfn). En caso de que se elija un número superior a 128, un símbolo de "line feed" será incorporado cada vez que se dé al <RETURN>. Podría ser que no siempre se desea esta característica por lo cual tendría que elegir números inferiores al 128.

El número de fichero tiene a continuación un número de aparato (2), que, en el caso del RS-232 será siempre el número 2. Esta es la señal que indica al programa que se está abriendo un canal RS-232. A continuación tenemos el 0 que es simplemente un número que guarda esta posición. La parte fundamental y más importante en el proceso de colocar los parámetros viene en las dos próximas sentencias —los registros de control y comando.

Registro de Control

El registro de control consiste en un registro de un byte (8 bits) que almacena varios de los parámetros necesarios para la utilización de un canal RS-232. Existe una imagen del registro de control que reside en la localización de memoria 659 decimal.

Los chips utilizados para la salida RS-232 en el VIC-20 no son iguales a los del C-64. El VIC-20 utiliza el VIA (adaptador de interface versatil), un chip 6522, mientras que el C-64 utiliza un CIA (adaptador de interface complejo), un chip 6526. Los registros de control y comando viven dentro de estos chips, pero para propósitos de programación, ponemos una imagen duplicada de los registros de los chips en las tarjetas dentro de la memoria para que su acceso sea más fácil. En estos 8 bits tenemos almacenadas tres piezas importantes de información.

Velocidad

En primer lugar habremos de definir el valor de los baudios (la velocidad de transmisión), lo que podemos considerar como el control de velocidad del canal RS-232. Esto es un velocímetro que indica al ordenador la rapidez con que ha de mandar y recibir la información que pasa a través del canal RS-232.

El control de velocidad se asigna en los primeros cuatro bits del registro de control. No hay que olvidar que al contar los bits, siempre se comienza en el 0 del bit que esté más a la derecha, yendo hacia la izquierda. Los programas de software del ordenador ya tienen incorporados los valores de baudios más comunes. La mayoría de vosotros ya estaréis familiarizados con los 300 baudios, una velocidad muy común en las comunicaciones vía modem, y con los 1.200 baudios, otra ▶

velocidad común para la transmisión de datos.

Pero antes de continuar, definamos un baudio. El valor de baudios es el número total de bits que se mandan en un segundo. Este término se abrevia a bps. Los códigos ASCII existen en versiones de 8 bits y 7 bits.

Suponiendo que se utilice el ASCII standard de siete bits; cada símbolo que se transmita consistirá en 7 bits, más uno o dos bits de parada, lo que viene a decir una transmisión de 400 palabras por minuto a una velocidad de 300 baudios. A una velocidad de 1.200 baudios se pueden transmitir casi tres veces esta cifra de palabras.

A medida que incrementa la velocidad, también lo hace la anchura de banda de la señal de datos. Esto requiere más electrónica para recobrar la señal. También implica la necesidad para una unión de más calidad entre el ordenador y el aparato con el que se esté comunicando. A menudo aparecen problemas cuando se utiliza una línea telefónica standard.

Para darte una idea de las diferentes anchuras de banda, consideremos que una señal en código morse, tecleada lentamente (a una velocidad de 10 palabras por minuto) tiene una anchura de banda teórica de unos pocos ciclos mientras que el equivalente digital tiene una anchura de banda de unos cientos de miles de ciclos.

La tabla 1 muestra los baudios implementados en el VIC-20 y del C-64. Nota que aunque se hayan puesto velocidades de más de 2.400 baudios, estas no son asequibles por una configuración normal del VIC-20 o del C-64.

El bit 4 (el quinto bit) no se utiliza. Los bits 5 y 6 definen el tamaño de la palabra. Las transmisiones RS-232 son seriales. Es decir, que se manda un dato tras otro. (El otro método de comunicación es paralelo).

Se seleccionan tamaños de palabra de 5, 6, 7 u 8 bits. La tabla 2 nos muestra las posibles opciones.

En el último bit, el séptimo, se indica si se desea utilizar uno o dos bits de parada. Un 0 indica un bit de parada y un 1 indica dos bits de parada. Esta es la señal de sincronización entre dos aparatos de manera que sepamos cuándo un carácter termina y comienza el próximo.

También existen bits de comienzo. Cada carácter que se transmite contiene un bit de comienzo incorporado. Un bit de comienzo es standard y por lo tanto no hace falta decirle al ordenador cuántos hay que mandar. (Esta función viene preprogramada.)

El origen de los bits de comienzo y parada es anterior a la segunda guerra mundial, cuando todos los medios de transmisión de datos eran mecánicos y funcionaban a base de varios engranajes. Debido a la falta de eficacia de estos aparatos, era necesario mandar señales de sincronización al principio y al final de cada carácter.

Con nuestra tecnología actual no hace falta utilizar los bits de parada y comienzo ya que tu ordenador es capaz de sincronizarse por sí solo. Sin embargo, casi todos los métodos actuales de transmisión utilizan un bit de comienzo y uno o más de parada.



Los otros parámetros que controlarás, están dentro del registro de comando.

Registro de Comando

Una vez vistos cómo los baudios, tamaño de palabras y bits de parada son asignados dentro del registro de control; veremos ahora lo que podemos hacer utilizando el registro de comando.

En realidad no se ha de hacer nada con el registro de comando ya que Commodore convenientemente ha puesto todos sus valores (todos ceros) para el registro de manera que concuerdan con la mayoría de los requisitos necesarios para la comunicación de datos. Al igual que en el registro de control, en este registro también existe una imagen de un byte del registro presente en la dirección decimal 660.

Consideremos cómo se comunican entre sí los ordenadores. Imagínate a dos personas, una habla rápidamente y la otra persona habla como un disco de 45 rpm. que se esté tocando a 33 rpm. Aunque hablen el mismo lenguaje, la persona que habla lentamente tendrá dificultades en entender a la que hable rápidamente. Por lo tanto llegan a un acuerdo en que la persona más lenta dará la mano a la persona que hable aprisa cuando esté preparada a escuchar la próxima frase. Un cero en el bit 0 en la dirección 660 indica darse la mano por tres frases, y un uno indica darse la mano en x-frases.

En realidad los ordenadores Commodore no son capaces de realizar una verdadera comunicación CS-232, pero esta limitación se solventa fácilmente.

Los bits 1, 2 y 3 no se utilizan. El bit 4 es importante, ya que se decide a través de este si el "dúplex" deberá de ser un dúplex entero o un semi dúplex. La comunicación normal de datos permite una transmisión simultánea de datos en dos sentidos a la que se suele llamar una operación dúplex entera. Si sólo transmite un aparato a la vez, entonces tenemos una operación semi dúplex.

La forma de implementar un dúplex entero en su ordenador es poniendo el bit 4 en cero. Un uno indicaría que se desea una operación semi duplex.

Consideremos la paridad. La paridad, o su ausencia, se logra mediante los bits 5, 6 y 7. La tabla 3 nos muestra las opciones disponibles. Los chequeos de paridad son uno de los métodos utilizados para asegurarnos de que la información transmitida se reciba correctamente y sin errores.

Se ejecuta un simple chequeo en los datos y se calcula si se tiene un número par o impar de binarios 1 en cada palabra. Si utilizamos una paridad par, entonces el número total de bits debería de ser un número par. Si no lo es, entonces cambiamos el bit de paridad a 1 de modo que el total resulte ser par. La paridad impar sería el mismo proceso sólo que el total habrá de ser impar.

El comando OPEN necesitará una explicación algo más detallada. Para ello utilizaremos un ejemplo:

```
OPEN 3,2,0,CHR$(6+32)+CHR$(32+128)
```

¿De dónde ha salido el CHR$? ¡No te asustes! Como ya hemos comentado, se ha abierto un fichero (el 3), con un aparato RS-232 (el 2) y en el canal cero (el 0). Hasta ahora bien. La siguiente parte de la sentencia deberá ser la de los valores del registro de control.

El CHR$ (6+32) quiere decir que se desea poner los bits en el registro de control que correspondan a los decimales 6 y 32. La conversión del 6 al binario de 0110 (bits 1 y 2). El decimal 32 se convierte en el binario 100000 (bit número 5). Si miras las tablas verás que cuando se elijan los bits 1 y 2 se está indicando una operación de 300 baudios. El bit 5 por sí solo indica que se utilizarán palabras de siete bits. Esto ya comienza a parecerse a un programa para un modem!

A continuación tenemos CHR$ (32+128). Esta es la información para el registro de comando. Siguiendo el mismo proceso anterior, el 32 pasa a ser el número binario 100000 (bit 5) y el 128 pasa a ser el número binario 10000000 (bit 7). Si miras la tabla 3, verás cómo esta combinación

indica la marca transmitida y que se anula el chequeo de paridad.

Llegados a este punto, tan solo te harán falta unos cuantos comandos básicos antes de que estés listo para escribir tu software de terminal que necesitarás para comunicarte a través de la salida RS-232.

Los demás comandos asociados con el canal RS-232 (CLOSE, CMD, INPUT #, GET #, y PRINT #) funcionan todos de manera parecida a como funcionan con cualquier fichero.

El programa del listado 1 está diseñado para ser una ayuda en el aprendizaje del uso de la sentencia OPEN con la salida RS-232. Cuando el programa esté cargado, podrás ver el chip de control y los registros de comando cuando se establezcan para unos parámetros que varíen.

Para mantener el programa lo más sencillo posible, pon la sentencia OPEN en la línea 10. Si ejecuta el programa respondiendo a las preguntas de direción con 659 y 660, podrás ver los registros. Este programa puede utilizarse para ver cualquier dirección en tu ordenador por lo que podría ser interesante guardarlo como programa de utilidad.



TABLA 2

6	5	Tamaño palabra
0	0	8 Bit
0	1	7 Bit
1	0	6 Bit
1	1	5 Bit

Bit 4 No se utiliza

Bit 7 0= 1 bit parada
1= 2 bit parada

OPCIONES PARA EL TAMAÑO DE PALABRAS

TABLA 1

3	2	1	0	Velocidad baudios	
0	0	0	0	Usuario	
0	0	0	1	50	baud
0	0	1	0	75	baud
0	0	1	1	110	baud
0	1	0	0	134,5	baud
0	1	0	1	150	baud
0	1	1	0	300	baud
0	1	1	1	600	baud
1	0	0	0	1200	baud
1	0	0	1	1800	baud
1	0	1	0	2400	baud
1	0	1	1	3600	baud
1	1	0	0	4800	baud
1	1	0	1	7200	baud
1	1	1	0	9600	baud
1	1	1	1	19200	baud

BAUDIOS IMPLEMENTADOS EN EL VIC-20 Y C-64

Bit 0 — Si es cero: Tres frases
Si es uno: X frases

Bits 1, 2 y 3 no se utilizan

Bit 4 — Si es cero ⇔ se implementa operación dúplex
Si en 1 ⇔ operación medio dúplex.

TABLA 3

7	6	5	PARIDAD
0	0	0	Paridad anulada
0	0	1	Paridad impar
0	1	0	No definida
0	1	1	Paridad par.
1	0	0	No definida
1	0	1	Marca transmitida, paridad anulada
1	1	0	No definida
1	1	1	Espacio transmitido, paridad anulada

El conector de 24 agujas tal como el Sullins 06SUL1224E5, se puede obtener Priority One Electronics, 9161 Deering Ave., Chatsworth, CA 91311.

OPCIONES DE PARIDAD

```
5 REM INSERTAR LA SENTENCIA OPEN EN
  LA LINEA 10.
10 OPEN 3,2,0,CHR$(6+32)+CHR$(32+128)
20 REM SETS SCREEN FOR C-64 ONLY:POKE
   53280,1:POKE 5328 1,1:PRINT" (CTRL 1)"
30 PRINT "{SHFT CLR} {CRSR DN} REGISTER
   PEEKER"
40 PRINT "{CRSR DN} ENTER THE ADDRESS"
42 PRINT "(IN DECIMAL)"
45 PRINT "OF THE REGISTER"
50 PRINT "YOU WISH TO VIEW"
60 INPUT R
70 I=PEEK(R):PRINT "{CRSR DN} THE CU-
   RRENT DECIMAL":PRINT "CONTENT IS:";I
80 GOSUB 500
90 PRINT "{CRSR DN} THE BINARY LAYOUT
   IS:"
100 PRINT:PRINT R7$+R6$+R5$+R4$+R3$+R2$+
    R1$+R0$
110 PRINT "{CRSR DN} DO YOU WISH TO
    VIEW"
130 PRINT "ANOTHER REGISTER (Y/N)"
140 GET A$:IF A$="" THEN 140
150 IF A$<>"Y" THEN END
160 GOTO 30
500 R0$="0":R1$="0":R2$="0":R3$="0":R4$="0":
    R5$="0":R6$="0":R7$="0"
505 IF I AND 1 THEN R0$="1"
510 IF I AND 2 THEN R1$="1"
520 IF I AND 4 THEN R2$="1"
530 IF I AND 8 THEN R3$="1"
540 IF I AND 16 THEN R4$="1"
550 IF I AND 32 THEN R5$="1"
560 IF I AND 64 THEN R6$="1"
570 IF I AND 128 THEN R7$="1"
580 RETURN
600 REM   ********************
610 REM   *
620 REM   *
630 REM   *    JIM GRUBBS
640 REM   *    PO BOX 3042
650 REM   *    SPRINGFIELD
660 REM   *    IL.62708
670 REM   *
680 REM   ********************
```

¿Cómo utilizar la sentencia OPEN con una salida RS-232

MUSIC 64

PERSONAL COMPUTER MUSIC





MUSICA CON COMPUTADOR con el Teclado "MUSIC 64"

El teclado "MUSIC 64" se conecta directamente al COMMODORE 64.

Se puede emplear como sintetizador monofónico o bien como teclado de órgano polifónico. Para ello se necesita lo siguiente:

- Un COMMODORE 64.
- Un vídeo monitor o un receptor de televisión.
- Y una unidad FLOPPY DISK o de cassettes.

ALBAREDA le proporciona a usted:

- Un teclado de 4 octavas de Do a Do.
- Un adaptador para realizar el empalme del teclado con el computador.
- El software necesario.

EL SINTETIZADOR MONOFONICO

La relación de los instrumentos de los cuales se dispone, está indicada en el vídeo monitor y son los siguientes:

0	Trompeta	7	Instrumentos de cuerda (cuerdas)
1	Brass		
2	Clarinete	8	Piano
3	Campanas	9	Organo eléctrico I
4	Flauta	A	Organo eléctrico II
5	Guitarra	B	Acordeón
6	Wha-Bras (cobreswha)	C	Random (sintetizador)

Modificaciones paramétricas

F1 para alcanzar el parámetro siguiente.
F3 para regresar al parámetro precedente.
F5 para aumentar el valor del parámetro elegido.
F7 para disminuir el valor del parámetro elegido.

EL TECLADO DEL ORGANO POLIFONO

La relación de los instrumentos de los cuales se dispone, está indicada en el vídeo monitor y son los siguientes:

1	Spinete	4	Flauta
2	Acordeón	5	Banjo
3	Campanas tubulares		

Modificaciones paramétricas

F1	Ataque	F2	Volumen
F3	Declive	F4	Traslado
F5	Sostener	F6	Forma de la onda
F7	Soltar	F8	Ciclo de la operación

Deseo recibir información completa del MUSIC-64 sin ningún compromiso por mi parte:

Nombre ....

Dirección ....

Teléf.: .... Ciudad ....

C.P. .... Provincia ....

Soy distribuidor... ☐ Soy particular... ☐

(*) Marcar con un asterisco lo que interese.



INSTRUMENTOS Y ACCESORIOS MUSICALES

C/. Carmen, 19
TARREGA (Lérida) - Teléfonos (973) 31 04 02 - 31 23 51

Por David D. BUSCH (RUN. EE.UU.)
Traducido por Valerie SHANKS y adaptado por Alvaro IBAÑEZ

Gran programa geográfico

GEOGRAFIA A JUEGO



Viaja, juega y aprende a la vez. ¿Quién da más? Echamos un cable a los chavales (que a veces la geografía resulta un poco pesada) y a los padres (que más de un patinazo se habrán pegado).

Esto es un juego, que no se equivoque nadie, pero de paso servirá para aprender capitales, ciudades, provincias, etc. Hay dos listados, uno para el C-64 y otro para el VIC-20, y otras líneas Data que van aparte que sirven para los dos. Las líneas Data en el listado para el VIC-20 (provincias de España) sirven también para el C-64, y las del listado para el C-64 (capitales y ciudades de Europa) sirven para el VIC-20. En el listado para las provincias españolas tienes que identificar las provincias que corresponden a los lugares que descienden por la pantalla. En el listado para capitales y ciudades de Europa, tienes que identificar el país. Las líneas Data corresponden al Continente Americano, en que de nuevo, tú tienes que identificar la capital o ciudad importante y teclear el nombre del país.

Bueno, ahora vamos a jugar para que puedas ver cómo funciona esto en la práctica. Como esto tiene que ver con ciudades, lo que aparece en la parte inferior de la pantalla es una línea de rascacielos y edificios grandes. Los nombres de los lugares empiezan a caer de tres en tres a diferentes velocidades y si uno llega a tocar los rascacielos pierdes una vida. Si esto ocurre tres veces perderás el juego. Para evitar que un lugar toque el suelo tienes que teclear el nombre de la provincia o país (según el listado que estás usando) más Return. Si aciertas, el nombre que has eliminado sera sustituido por otro, y los tres empezarán de nuevo a descender por la pantalla. Si eliminas 25, ganas el juego.

C-64
VIC-20 + 3K

Las preguntas y respuestas utilizadas en este programa se almacenan de dos en dos en un "string" de "arrays". En este caso, el "array" se llama GEO$ (ROW,-COL), donde se carga.

Control del cursor

Las preguntas y respuestas utilizadas en este programa se almacenan de dos en dos en un "string" de "arrays". En esta caso, el "array" se llama GEO$(ROW,-COL), donde se carga cada país/provincia/ciudad (por ejemplo, Portugal-Lisboa, Vizcaya-Bilbao).

En la presentación de las preguntas, se han definido tres columnas en la pantalla, en las columnas, 1, 15 y 25 (las columnas 1, 6 y 11 en el VIC-20). Los nombres de los lugares no salen de estas columnas, aunque las filas donde aparecen aumentarán a medida que van bajando por la pantalla.

Se ha empleado un método bastante interesante para que los nombres de los lugares aparezcan en pantalla. Tanto el C-64 como el VIC-20 requieren unas operaciones bastante complicadas para realizar las funciones que otros ordenadores llevan incorporadas.

Por ejemplo, algunos ordenadores te permiten desplazar el cursor a una posición determinada para empezar a escribir el siguiente carácter en la misma posición mediante una sentencia dada, como puede ser PRINT @ o LOCATE. Se prefiere esta última porque el comando LOCATE te permite especificar la fila y la columna donde quieres situar el cursor.

Con los primeros ordenadores Radio Shack, los IBM y hasta los Spectrum, puedes entrar PRINT @ en cualquiera de más de 1000 posiciones de pantalla. Por ejemplo:

```
10 PRINT @ 0, "HOLA"
20 PRINT @ 64, "HOLA"
```

De esta forma se escribe HOLA en la fila 1, columna 1 y la fila 2, columna 1 de la pantalla. Esto no es lo ideal, porque tienes que calcular cada posición inicial en la pantalla. Podrías comparar esto con la operación de introducir un carácter mediante un Poke en la memoria de pantalla en los ordenadores Commodore. Por ejemplo:

```
10 POKE P,1
20 POKE P+40,1
```

Si P hubiera sido definido previamente en el programa como el principio de la memoria de pantalla, la letra A se hubiera introducido mediante un Poke en la fila 1, columna 1, y fila 2, columna 2 (solamente en el C-64 sustituye P+22 para el VIC-20).

Ten en cuenta que tenías que calcular que la fila 2, columna 2, era P+40 para que la letra se situara en el lugar correcto. También toma nota que resulta más difícil escribir en las posiciones de pantalla con los ordenadores Commodore dado que los caracteres se tienen que introducir uno por uno mediante un Poke, y tienes que hacer un Poke en la memoria de color con el caracter de color. Finalmente, tienes que tener en cuenta que los códigos de los Poke en los ordenadores Commodore no son iguales que los códigos CHR$. La A es CHR$(65), pero tiene un valor de 1 cuando se hace un Poke.

Algunos ordenadores, como los IBM, utilizan el método de fila y columna para el direccionamiento de la pantalla. Por ejemplo:

```
LOCATE 1,1: PRINT"HOLA"
LOCATE 2,1: PRINT"HOLA"
```

Esto funciona igual que el primer ejemplo, excepto que no hace falta calcular las posiciones de pantalla. Esta característica resulta muy útil para el desplazamiento de objetos en pantalla, dado que se pueden utilizar variables, como ROW y COL, y cambiarlas. Si quieres que el objeto se desplace hacia la derecha, haces que COL=COL+1. Para que se desplace hacia la izquierda, haces que COL=COL-1. Para ir hacia abajo, ROW=ROW+1. ¿Vas cogiendo la idea?

Este programa añade esta característica al C-64 y al VIC-20. El ordenador dispone de una característica muy útil que es el control del cursor con el que puedes desplazar el cursor por filas y por columnas en cualquier dirección. Para poder imitar la función Locate, tienes que hacer lo siguiente:



—Elegir la fila y columna donde quieres que aparezca el cursor.
—Desplazar el cursor a la posición HOME. Esto se hace escribiendo CHR$(19).
—Desplazar los caracteres de cursor COL a la derecha.
—Desplaza el cursor hacia abajo el número ROM de filas.
—Empieza a escribir en este punto.

Esto es mucho más fácil de lo que parece. Iré despacio para que puedas seguir los pasos, utilizando la pantalla de 40 columnas del C-64 como ejemplo.

Como funciona

Primero, se construye un "string" que devuelve el cursor a la posición HOME, sin borrar la pantalla. Este "string" también incluye suficientes símbolos de CRSR RT para desplazar el cursor a la extrema derecha de una sola fila. Por lo tanto, esta variable, R$, será igual a 1 CHR$(19)(HOME) más 40 CHR$(29) (CRSR RT). Esto se construye de la siguiente forma:

```
10 R$=CHR$(19)
20 FOR N=1 TO 40
30 R$=R$+CHR$(29)
40 NEXT N
```

Si se incluye PRINT R$ en cualquier lugar del programa, el cursor aparecerá en seguida en la esquina superior izquierda de la pantalla, sin borrar lo que se encuentra allí, y a continuación se desplazará 40 posiciones a la derecha. Esto ocurre tan de prisa que ni te das cuenta de ello.

Primero, se construye un "string" que devuelve el cursor a la posición HOME, sin borrar la pantalla. Este "string" también incluye suficientes símbolos de CRSR RT para desplazar el cursor.

Si no quieres que el cursor esté en la columna 40 o si COL es igual a 20 tienes que utilizar LEFT$. Por ejemplo:

```
100 PRINT LEFT$,(R$,COL):"HOLA"
```

Ahora presta atención. El cursor se desplaza a la esquina superior izquierda, igual que antes porque el primer carácter de R$ sigue siendo CHR$(19). Pero en este caso el C-64 escribe los siguientes caracteres solamente hasta la longitud de COL-1, como si fueran de CRSR RT. Por lo tanto el cursor se desplaza un total de COL-1 espacios a la derecha, y empieza a escribir en la siguiente posición, que es COL.

Por lo tanto, si utilizas la línea 100 que se ha visto antes, puedes escribir HOLA en la fila 1 y el número de columna COL casi instantáneamente. Pero puede que quieras escribir en una fila que no sea la 1.

Para hacer esto se construye una segunda variable, D$, que es igual a 25 símbolos de CRSR DN, o CHR$(17). Si quieres la fila 10, ROW tiene que ser igual a 10. La línea del programa serán las siguientes:

```
100 PRINT LEFT$(R$,COL);LEFT$
    (D$,ROW);"HOLA"
```

El cursor se desplazará a la posición HOME, pasar por las columnas COL y hacia abajo en las filas ROW, para que puedas escribir automáticamente en ROW, COL. Enhorabuena! Acabas de añadir la función LOCATE.

Esta técnica funciona exactamente igual que en el VIC-20, con la diferencia de que el cursor solamente se tiene que desplazar por 22 columnas para cubrir toda la pantalla de izquierda a derecha.

Esta técnica se utiliza en este programa. Se eligen los nombres de tres lugares accediendo tres veces a una subrutina en la línea 770 (la línea 780 en el listado del VIC-20). Esta rutina elige un número aleatorio, R, y, si dicho lugar no ha sido escogido antes, GEØ$(R,1) se convierte en un nombre de lugar, CIUDAD$(n,1). La respuesta que se necesita para eliminar el lugar, GEO$(R,2), se almacena en CIUDAD$(n,2).

La fila donde se escribe cada ciudad en un momento dado se almacena en un "array" numérico de tres elementos, uno para cada una de las tres ciudades. Al principio de cada vuelta, R(1), R(2) y R(3) serán todos iguales a 1, dado que las ciudades 1, 2 y 3 empezarán en la fila 1.

A medida que avanza el juego, la sentencia LOCATE se utiliza para situar el cursor en la columna y fila donde tiene que aparecer una ciudad. Para escribir la ciudad Q, utiliza las siguientes líneas.

```
930 PRINT LEFT$(R$,COL(Q));
940 PRINT LEFT$(D$,R(Q));
950 FOR S=1 TO LEN(CIUDAD
    $(Q,1))
960 PRINT CHR$(32);
```

```
970 NEXT S
980 R(Q)=R(Q)+1
990 PRINT LEFT$(R$,COL(Q));
    LEFT$(D$,R(Q));
1000 PRINT CIUDAD$(Q,1);
```

Como te acordarás, COL(n), siempre será igual para cada ciudad. Por lo tanto, la línea 930 coloca el cursor en la columna fijada para el lugar Q y la línea 940 desplaza el cursor a la fila correcta.

A continuación, un bucle For...Next, desde 1 hasta donde acabe el nombre de la ciudad, escribe una serie de espacios, CHR$(32), en la pantalla, para borrar el nombre de la ciudad en la última posición donde apareció. Luego, se suma 1 a la fila donde la ciudad Q aparece y se escribe el nombre de la ciudad en la nueva posición.

Dado que Q se elige al azar (línea 900 en el listado para el C-64, 910 en el del VIC-20), a veces una ciudad se desplazará varias veces en una fila mientras las demás se quedan quietas. Otras veces los movimientos serán iguales. Esto resulta en el descenso aleatorio de las tres palabras.

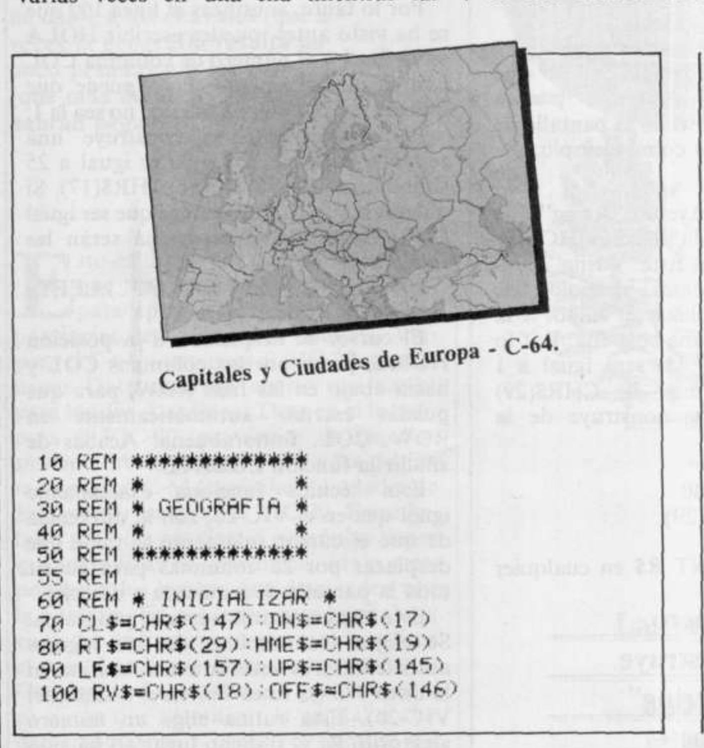
Si cualquiera de las tres variables R(n) es igual al número de líneas que representa la capacidad máxima de la pantalla del C-64 o la del VIC-20, significa que la ciudad ha llegado a la última fila de la pantalla, y el programa accede a una subrutina que resta una vida y avisa con un sonido y escribe una advertencia. También comprueba para ver si LIVES=0, y, si es así, se termina el juego.

A medida que van bajando las ciudades, el teclado busca una entrada (líneas 910 y 920). Si pulsas una tecla, el programa salta a la línea 1150 o 1160. Aquí, se hace una comprobación para ver si A$ corresponde a RETURN (CHR$(13)). De ser así, el programa vuelve a trasladarse, en esta ocasión a las líneas 1250-1260, para ver si la respuesta, AN$, corresponde a cualquiera de las ciudades en la pantalla en ese momento.

Al recibir una respuesta correcta se borran las posiciones anteriores de las ciudades, sus filas vuelven a tener un valor 1 y se elige un nuevo lugar para sustituir al anterior.

A veces, los nombres de las ciudades chocan en la pantalla, especialmente en la pantalla del VIC-20 por ser esta más pequeña. Sin embargo, como se empiezan a escribir unos encima de otros, resulta más divertido al ser más difícil reconocerlos.

Si quieres modificar el color de los caracteres, cambia el valor de la variable CYAN$ en la línea 110. NC en la línea 130 es el número de parejas de ciudades. Si quieres aumentar o reducir este número (siempre dejando un mínimo de 28) tienes que modificarlo en esta línea.



Capitales y Ciudades de Europa - C-64.

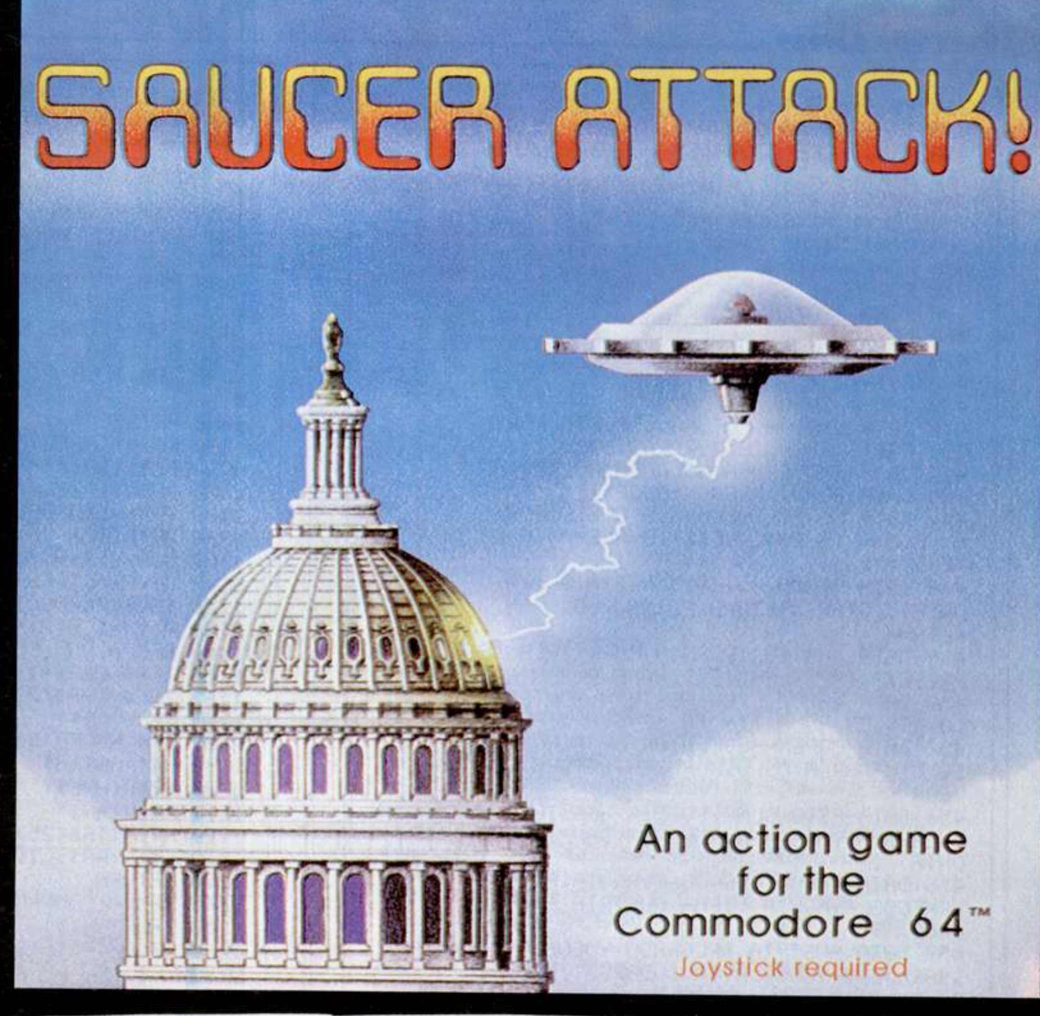
```
10 REM ***************
20 REM *             *
30 REM * GEOGRAFIA *
40 REM *             *
50 REM ***************
55 REM
60 REM * INICIALIZAR *
70 CL$=CHR$(147):DN$=CHR$(17)
80 RT$=CHR$(29):HME$=CHR$(19)
90 LF$=CHR$(157):UP$=CHR$(145)
100 RV$=CHR$(18):OFF$=CHR$(146)
110 WHITE$=CHR$(5):CYAN$=CHR$(159)
120 GOSUB1580
130 NC=65
140 LIVES=3
150 REM * INSTRUCCIONES *
160 PRINTCL$;DN$
170 PRINTTAB(6)"[5SPC]*[3SPC]GEOGRAF
IA[2SPC]*"
180 PRINT
190 PRINTTAB(4)"ENTRA[SPC]EL[SPC]PAI
S[SPC]O[SPC]LA[SPC]PROVINCIA"
200 PRINTTAB(4)"EN[SPC]LA[SPC]QUE[SPC]
SE[SPC]ENCUENTRA[SPC]LA[SPC]CIUDAD"
210 PRINT
220 PRINTTAB(4)"TIENES[SPC]TRES[SPC]
VIDAS[SPC]PARA[SPC]ELIMINAR"
230 PRINTTAB(4)"25[SPC]CIUDADES.[SPC]
BUENA[SPC]SUERTE!";DN$
240 PRINTTAB(10)"[7SPC]PULSA[SPC]UNA
[SPC]TECLA"
250 GETA$:IFA$=""GOTO250
260 REM * DIFICULTAD *
270 PRINTCL$;DN$;DN$
280 PRINTTAB(8)"DIFICULTAD:";DN$
290 PRINTTAB(8)"[1][SPC]RAPIDO[SPC]A
[SPC][9][SPC]LENTO"
300 GETA$:IFA$=""THEN300
```

¡POR FIN!

Ha llegado un Nº 1 en USA para su

Commodore 64

Ud. está en su posición vigilando Washington D.C.
La ciudad parece apacible desde su puesto de defensa y no hay los signos de la invasión Extraterrestre que Ud. espera.
De pronto el silencio es alterado por el penetrante silbido de los platillos volantes. Son los invasores descendiendo sobre la ciudad.
Ud. entra en acción y pone en su punto de mira al platillo volante.
Aprieta el botón de juego y sale un misil que destruye al ovni desintegrándose el platillo. Otro platillo volante se acerca a los edificios y antes de que Ud. se de cuenta destruye el monumento de Washington C.D. con un rayo de energía atómica. La batalla transcurre durante el día y en un momento determinado pasa a ser de noche.
Si un platillo aterriza todo está perdido. Cuando Ud. consiga neutralizar el ataque el juego continuará y se deberá destruir la nave madre. Si lo consigue una gran celebración le espera cuando vuelva a Washington D.C.



SAUCER ATTACK *es realmente un* JUEGO DISTINTO



FOTOS
TOMADAS DIRECTAMENTE
DE UN MONITOR 1702
COMMODORE

INSUPERABLE CALIDAD AUDIO-VISUAL

PEDIDOS

FERRE-MORET S.A.

Tuset n.º 8, entlo. 2.º Tel. 218 02 93
BARCELONA 08006

DISTRIBUIDORES Y PARTICULARES

NOMBRE..
DIRECCION.......................................
CIUDAD........................ PROVINCIA ....

☐ Adjunto cheque
☐ Contra reembolso

Gastos envio: **300 ptas.**

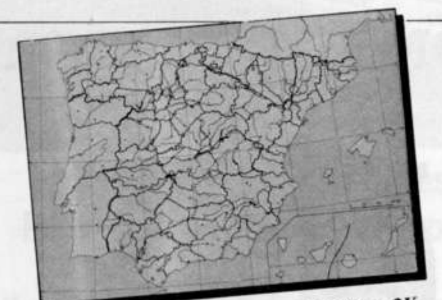
Precio Venta
9.975 ptas.

```
310 DELAY=VAL(A$)*2
320 IF DELAY=0GOTO300
330 REM * LEE CIUDADES Y LAS PONE EN
 EL ARRAY*
340 PRINTCL$
350 COL(1)=1:COL(2)=15:COL(3)=25
360 DIMGEO$(NC,2)
370 FORROW=1TONC
380 FORCOL=1TO2
390 READ GEO$(ROW,COL)
400 NEXTCOL
410 NEXTROW
420 PRINTCL$
430 REM * INFORMACION *
440 DATA LISBOA,POTUGAL,OPORTO,PORTU
GAL,PARIS,FRANCIA,BURDEOS,FRANCIA
442 DATA AMSTERDAM,HOLANDA,ROTTERDAM
,HOLANDA,BONN, ALEMANIA OESTE
444 DATA MUNICH,ALEMANIA OESTE,POSTD
AM,ALEMANIA ESTE,LEIPZIG,ALEMANIA ES
TE
446 DATA BERNA,SUIZA,GINEBRA,SUIZA,L
ONDRES,GRAN BRETANA,EDIMBURGO,GRAN B
RETANA
448 DATA DUBLIN,IRLANDA,CORK,IRLANDA
,REYKJAVIK,ISLANDIA,TIRANA,ALBANIA
450 DATA VADUZ,LIENCHESTEIN,MONTECAR
LO,MONACO,ROMA,ITALIA,GENOVA,ITALIA
452 DATA COPENHAGUE,DINAMARCA,ROSKIL
DE,DINAMARCA,OSLO,NORUEGA,BERGEN,NOR
UEGA
454 DATA ESTOCOLMO,SUECIA,HENSILKI,F
INLANDIA,VARSOVIA,POLONIA,DANZIG,POL
ONIA
456 DATA PRAGA,CHECOSLOVAQUIA,PILSEN
,CHECOSLOVAQUIA,VIENA,AUSTRIA,SALTZB
URGO
458 DATA AUSTRIA,BELGRADO,YUGOSLAVIA
,SARAJEVO,YUGOSLAVIA,BUDAPEST,HUNGRI
A
460 DATA SOFIA,BULGARIA,BUCAREST,RUM
ANIA,CONSTANZA,RUMANIA,ATENAS,GRECIA

462 DATA SALONICA,GRECIA,MOSCU,RUSIA
,ODESSA,RUSIA,RIGA,LETONIA,KIEV,UCRA
NIA
464 DATA ANKARA,TURQUIA,NICOSIA,CHIP
RE,LA VALETA,MALTA,SICILIA,ITALIA
466 DATA CORCEGA,FRANCIA,CERDENA,ITA
LIA,CRETA,GRECIA,CARDIFF,GRAN BRETAN
A
468 DATA GLASGOW,GRAN BRETANA,FRANKF
URT,ALEMANIA OESTE,COLONIA,ALEMANIA
OESTE
470 DATA MARSELLA,FRANCIA,LYON,FRANC
IA,LA HAYA,HOLANDA
472 DATA VILNA,LITUANIA,TALLIN,ESTON
IA,ESTAMBUL,TURQUIA,BRUSELAS,BELGICA

474 DATA AMBERES,BELGICA
760 GOTO810
770 R=INT(RND(1)*NC)+1
780 IFGEO$(R,1)=""GOTO770
790 RETURN
800 REM * COLOCA 3 INVASORES *
810 FORN=1TO3
820 GOSUB770
830 CIUDAD$(N,1)=GEO$(R,1):CIUDAD$(N
,2)=GEO$(R,2)
840 GEO$(R,1)=""
850 NEXTN
860 REM * COMIENZA EL JUEGO *
870 PRINTCL$
880 GOSUB1660
890 R(1)=1:R(2)=1:R(3)=1
900 Q=INT(RND(1)*3)+1
910 GETA$:IFA$<>""THENGOSUB1150
920 IFR(1)=25ORR(2)=25ORR(3)=25THENG
OSUB1050
930 PRINTLEFT$(R$,COL(Q));
940 PRINTLEFT$(D$,R(Q));
950 FORS=1TOLEN(CIUDAD$(Q,1))
960 PRINTCHR$(32);
970 NEXTS
980 R(Q)=R(Q)+1
990 PRINTLEFT$(R$,COL(Q));LEFT$(D$,R
(Q));
1000 PRINTCIUDAD$(Q,1);
1010 GOSUB1720
1020 FORN=1TODELAY:NEXTN
1030 GOTO900
1040 REM * LA CIUDAD LLEGA AL SUELO
*
1050 PRINTCL$
1060 PRINTTAB(2)"HAS[SPC]PERDIDO[SPC]
UNA[SPC]VIDA!!"
1070 GOSUB1820
1080 FORN=1TO1000:NEXTN
1090 PRINTCL$
1100 GOSUB1660
1110 LIVES=LIVES-1
1120 IFLIVES=0GOTO1430
1130 R(1)=1:R(2)=1:R(3)=1:GOTO1400
1140 REM * UNA LETRA *
1150 IFA$=CHR$(13)THEN1250
1160 IFA$=CHR$(32)THEN1190
1170 A=ASC(A$)
1180 IFA<65ORA>90THENRETURN
1190 AN$=AN$+A$
1200 PRINTHME$;
1210 PRINTAN$
1220 PRINTTAB(25)RV$;WHITE$;"VIDAS[SPC]
:";OFF$;CYAN$;LIVES;
1230 RETURN
1240 REM * COMPRUEBA RESPUESTA *
1250 IF RIGHT>13 THEN1500
1260 IF AN$=CIUDAD$(1,2)THENAN=1:GOT
O1330
1270 IF AN$=CIUDAD$(2,2)THENAN=2:GOT
O1330
1280 IF AN$=CIUDAD$(3,2)THENAN=3:GOT
O1330
1290 AN$=""
1300 PRINTHME$;"[17SPC]";:PRINTHME$
1310 RETURN
1320 REM * RESPUESTA CORRECTA *
1330 RIGHT=RIGHT+1
1340 PRINTCL$
1350 GOSUB1660
1360 GOSUB770
1370 CIUDAD$(AN,1)=GEO$(R,1):CIUDAD$
(AN,2)=GEO$(R,2)
1380 GEO$(R,1)=""
1390 R(1)=1:R(2)=1:R(3)=1
1400 AN$=""
1410 RETURN
1420 REM * SE PIERDE *
1430 PRINTCL$;DN$;DN$
1440 PRINTTAB(4)"LO[SPC]SIENTO[SPC],
[SPC]PERO[SPC]HAS[SPC]AGOTADO[SPC]TU
S"
1450 PRINTTAB(4)"TRES[SPC]VIDAS.RESP
ONDISTE";RIGHT
1460 PRINTTAB(4)"CORRECTAMENTE."
1470 PRINT
1480 GOTO1540
1490 REM * SE GANA *
1500 PRINTCL$;DN$;DN$
1510 PRINTTAB(10)"HAS[SPC]GANADO!"
1520 PRINT
1530 REM * FIN *
1540 PRINTTAB(10)"OTRA[SPC]VEZ?"
1550 GETA$:IFA$=""GOTO1550
1560 IFA$="S"THENRUN
1570 END
1580 R$=HME$
1590 FORCOL=1TO40
1600 R$=R$+RT$
```

```
1610 NEXTCOL
1620 FORROW=1TO25
1630 D$=D$+DN$
1640 NEXT ROW
1650 RETURN
1660 PRINTLEFT$(R$,1);LEFT$(D$,25);
1670 FORB=1TO39
1680 C=((INT(RND(1)*3)+1)*10)+152
1690 PRINTCHR$(C);
1700 NEXTB
1710 RETURN
1720 FORN=15 TO 14 STEP-1
1730 POKE54296,N
1740 POKE54276,129
1750 POKE54277,15
1760 POKE54273,40
1770 POKE54272,200
1780 NEXTN
1790 POKE54276,0
1800 POKE54277,0
1810 RETURN
1820 POKE54296,15
1830 POKE54276,33
1840 POKE54277,128
1850 POKE54278,128
1860 FORJ=1TO2
1870 FORN=1TO100
1880 POKE54273,N/2+30
1890 POKE54272,N/2+70
1900 NEXTN
1910 NEXTJ
1920 POKE54296,0
1930 RETURN
```



Provincias de España - VIC-20 + 3K.

```
10 REM **************
20 REM *            *
30 REM * GEOGRAFIA  *
40 REM *            *
50 REM **************
55 REM
60 REM * INICIALIZAR *
70 CL$=CHR$(147):DN$=CHR$(17)
80 RT$=CHR$(29):HME$=CHR$(19)
90 LF$=CHR$(157):UP$=CHR$(145)
100 RV$=CHR$(18):OFF$=CHR$(146)
110 WHITE$=CHR$(5):CYAN$=CHR$(159)
120 GOSUB1580
125 POKE36879,8
130 NC=65
140 LIVES=3
150 REM * INSTRUCCIONES *
160 PRINTCL$;DN$
170 PRINT"[YEL][SPC]*[4SPC]GEOGRAFIA
[4SPC]*";DN$
180 PRINT
190 PRINT"ENTRA[SPC]EL[SPC]PAIS[SPC]
O[SPC]LA[4SPC][CRSRD]PROVINCIA[SPC]D
ONDE[SPC]SE[4SPC][CRSRD]ENCUENTRA[SPC]
LA[SPC]CIUDAD."
200 PRINT"[CRSRD]TIENES[SPC]3[SPC]VI
DAS[SPC]PARA[3SPC][CRSRD]ELIMINAR[SPC]
25[SPC]CIUDADES.":PRINT"[2CRSRD]BUEN
A[SPC]SUERTE!"
210 PRINT
220 REM
230 REM
240 PRINT"[3SPC]PULSA[SPC]UNA[SPC]TECLA"
250 GETA$:IFA$=""GOTO250
260 REM * DIFICULTAD *
270 PRINTCL$;DN$;DN$
280 PRINT"DIFICULTAD?:";DN$
290 PRINT"[1][SPC]LENTO[SPC]A[SPC][9
][SPC]RAPIDO"
300 GETA$:IFA$=""THEN300
310 DELAY=VAL(A$)*2
320 IF DELAY=0GOTO300
330 REM * LEE CIUDADES Y LAS COLOCA
EN EL ARRAY *
340 PRINTCL$
350 COL(1)=1:COL(2)=8:COL(3)=15
360 DIMGEO$(NC,2)
370 FORROW=1TONC
380 FORCOL=1TO2
390 READ GEO$(ROW,COL)
400 NEXTCOL
410 NEXTROW
420 PRINTCL$
430 REM * INFORMACION *
440 DATA FERROL,LA CORUNA,VIGO,PONTE
VEDRA,RIBADEO,LUGO,RIBADAVIA,ORENSE
442 DATA OVIEDO,ASTURIAS,GIJON,ASTUR
IAS,TORRELAVEGA,SANTANDER,BILBAO,VIZCAYA
444 DATA S.SEBASTIAN,GUIPUZCOA,VITOR
IA,ALAVA,PAMPLONA,NAVARRA,MONZON,HUESCA
446 DATA CALATAYUD,ZARAGOZA,ALCANIZ,
TERUEL,SABADELL,BARCELONA,REUS,TARRAGONA
448 DATA OLOT,GERONA,SEO DE URGEL,LE
RIDA,CALAHORRA,LOGRONO,ELDA,ALICANTE
450 DATA SAGUNTO,VALENCIA,PENISCOLA,
CASTELLON,CARTAGENA,MURCIA,MANACOR,M
ALLORCA
452 DATA MAHON,MENORCA,PONFERRADA,LE
ON,BENAVENTE,ZAMORA,ALBA DE TORMES,S
ALAMANCA
454 DATA CIUDAD RODRIGO,SALAMANCA,MI
RANDA DE EBRO,BURGOS ,NUMANCIA,SORIA
456 DATA SEPULVEDA,SEGOVIA,AREVALO,A
VILA,TORDESILLAS,VALLADOLID
458 DATA VENTA DE BANOS,PALENCIA,ARA
NJUEZ,MADRID,TALAVERA,TOLEDO
458 DATA VENTA DE BANOS,PALENCIA,ARA
NJUEZ,MADRID,TALAVERA,TOLEDO
460 DATA VALDEPENAS,CIUDAD REAL,TARA
NCON,CUENCA,SIGUENZA,GUADALAJARA
462 DATA HELLIN,ALBACETE,PALOS DE MO
GUER,HUELVA,ECIJA,SEVILLA,CARMONA,SE
VILLA
464 DATA LUCENA,CORDOBA,BAENA,CORDOB
A,BAILEN,JAEN,MOJACAR,ALMERIA
466 DATA MOTRIL,GRANADA,ANTEQUERA,MA
LAGA,ALGECIRAS,CADIZ,TRUJILLO,CACERES
468 DATA MERIDA,BADAJOZ,DON BENITO,B
ADAJOZ,LA OROTAVA,TENERIFE
470 DATA MAS PALOMAS,GRAN CANARIA,AR
RECIFE,LANZAROTE
472 DATA TUDELA,NAVARRA,JACA,NAVARRA
,HARO,LA RIOJA,LORCA,MURCIA,INCA,BAL
EARES
474 DATA GUADIX,GRANADA,RONDA,MALAGA
,ROTA,CADIZ
760 GOTO810
770 R=INT(RND(1)*NC)+1
780 IFGEO$(R,1)=""GOTO770
790 RETURN
800 REM * COLOCA 3 CIUDADES *
810 FORN=1TO3
820 GOSUB770
830 CIUDAD$(N,1)=GEO$(R,1):CIUDAD$(N
,2)=GEO$(R,2)
840 GEO$(R,1)=""
850 NEXTN
860 REM * COMIENZA EL JUEGO *
870 PRINTCL$
```

Le presentamos a SPINNAKER.
Para que también nuestros hijos aprendan jugando con el ordenador.

Muchos padres ya conocen que el ordenador doméstico puede servir para que nuestros hijos jueguen y pasen largos ratos de ocio. Con SPINNAKER los **juegos de ordenador** además sirven **para aprender**.

SPINNAKER es la firma líder mundial en juegos educativos para ordenador. Ahora llega a nuestro país para que también nuestros hijos aprendan jugando con el ordenador.

Nuestros juegos están llenos de emoción, entretenimiento, color, música... Pero además poseen un alto grado de valor educativo, ayudando a los niños a que desarrollen sus habilidades y aprendan de una divertida manera. Aprovechando al máximo las nuevas posibilidades que nos ofrece el mundo de las computadoras.

En su elaboración han intervenido especialistas en la educación, la didáctica, la música, los gráficos y la programación, que han realizado un cuidado producto especialmente pensado para la educación.

Estos son los motivos por los que miles de niños de todo el mundo son verdaderos fans de SPINNAKER. Y sus padres también.

Si Ud. realmente desea que su ordenador familiar sea una ayuda útil para el futuro de sus hijos, recuerde este nombre: SPINNAKER: Una divertida familia de juegos educativos.



DISPONEMOS DE VERSIONES PARA APPLE II, Commodore 64 y Sinclair Spectrum.

Precio especial de introducción 2.400 pts.



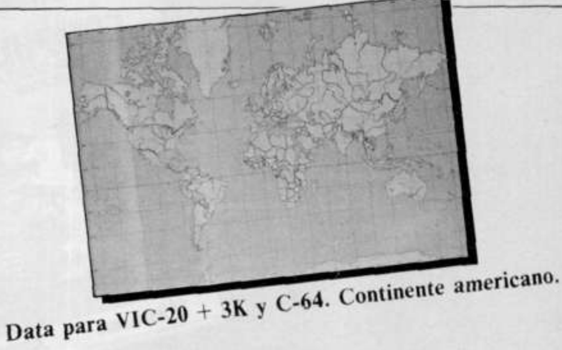
Gran Via Carlos III, 97 K. 08028 BARCELONA



SPINNAKER SOFT COR. CAMBRIDGE, MASS. USA.

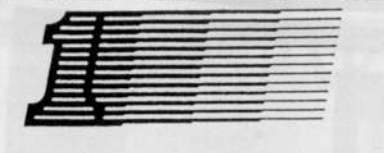
```
880 GOSUB1660
890 R(1)=1:R(2)=1:R(3)=1
900 Q=INT(RND(1)*3)+1
910 GETA$:IFA$<>""THENGOSUB1150
920 IFR(1)=23ORR(2)=23ORR(3)=23THENG
OSUB1050
930 PRINTLEFT$(R$,COL(Q));
940 PRINTLEFT$(D$,R(Q));
950 FORS=1TOLEN(CIUDAD$(Q,1))
960 PRINTCHR$(32);
970 NEXTS
980 R(Q)=R(Q)+1
990 PRINTLEFT$(R$,COL(Q));LEFT$(D$,R(Q));
1000 PRINTCIUDAD$(Q,1);
1010 GOSUB1720
1020 FORN=1TODELAY:NEXTN
1030 GOTO900
1040 REM * LA CIUDAD LLEGA AL SUELO *
1050 PRINTCL$
1060 PRINT"HAS[SPC]PERDIDO[SPC]UNA[SPC]
VIDA!"
1070 GOSUB1720
1080 FORN=1TO1000:NEXTN
1090 PRINTCL$
1100 GOSUB1660
1110 LIVES=LIVES-1
1120 IFLIVES=0GOTO1430
1130 R(1)=1:R(2)=1:R(3)=1:GOTO1400
1140 REM * UNA LETRA *
1150 IFA$=CHR$(13)THEN1250
1160 IFA$=CHR$(32)THEN1190
1170 A=ASC(A$)
1180 IFA<65ORA>90THENRETURN
1190 AN$=AN$+A$
1200 PRINTHME$;
1210 PRINTAN$
1220 PRINTTAB(12)RV$;WHITE$;"VIDAS[SPC]
:";OFF$;CYAN$;LIVES;
1230 RETURN
1240 REM * COMPRUEBA RESPUESTA *
1250 IF RIGHT>13 THEN1500
1260 IF AN$=CIUDAD$(1,2)THENAN=1:GOT
O1330
1270 IF AN$=CIUDAD$(2,2)THENAN=2:GOT
O1330
1280 IF AN$=CIUDAD$(3,2)THENAN=3:GOT
O1330
1290 AN$=""
1300 PRINTHME$;"[17SPC]";:PRINTHME$
1310 RETURN
1320 REM * RESPUESTA CORRECTA *
1330 RIGHT=RIGHT+1
1340 PRINTCL$
1350 GOSUB1660
1360 GOSUB770
1370 CIUDAD$(AN,1)=GEO$(R,1):CIUDAD$
(AN,2)=GEO$(R,2)
1380 GEO$(R,1)=""
1390 R(1)=1:R(2)=1:R(3)=1
1400 AN$=""
1410 RETURN
1420 REM * SE PIERDE *
1430 PRINTCL$;DN$;DN$
1440 PRINT"PERDONA,PERO[SPC]HAS[6SPC]
[CRSRD]PERDIDO"
1450 PRINT"[CRSRD]TRES[SPC]VIDAS.ACE
RTASTE";RIGHT
1460 PRINTTAB(4)"CIUDADES"
1470 PRINT
1480 GOTO1540
1490 REM * SE GANA *
1500 PRINTCL$;DN$;DN$
1510 PRINTTAB(10)"HAS[SPC]GANADO[SPC]
!!"
1520 PRINT
1530 REM * JUEGAS OTRA VEZ *
1540 PRINTTAB(10)"OTRA[SPC]VEZ?"
1550 GETA$:IFA$=""GOTO1550
1560 IFA$="S"THENRUN
1570 END
```

```
1580 R$=HME$
1590 FORCOL=1TO40
1600 R$=R$+RT$
1610 NEXTCOL
1620 FORROW=1TO25
1630 D$=D$+DN$
1640 NEXT ROW
1650 RETURN
1660 PRINTLEFT$(R$,1);LEFT$(D$,25);
1670 FORB=1TO21
1680 C=((INT(RND(1)*3)+1)*10)+152
1690 PRINTCHR$(C);
1700 NEXTB
1710 RETURN
1720 POKE36878,15
1730 POKE36874,230
1740 POKE36878,0
1750 RETURN
```



Data para VIC-20 + 3K y C-64. Continente americano.

```
440 DATA CIUDAD DE MEXICO,MEXICO,ACA
PULCO,MEXICO,GUADALAJARA,MEXICO
442 DATA TEGUCIGALPA,HONDURAS,SAN SA
LVADOR,EL SALVADOR,MANAGUA,NICARAGUA
444 DATA SAN JOSE,COSTA RICA,PANAMA,
PANAMA,KINGSTOWN,JAMAICA,LA HABANA,CUBA
446 DATA NASSAU,BAHAMAS,PUERTO PRINC
IPE,HAITI,SANTO DOMINGO,REP.DOMINICANA
448 DATA SAN JUAN,PUERTO RICO,BRIDGE
TOWN,BARBADOS,SAINT GEORGE'S,GRANADA
450 DATA PUERTO ESPANA,TRINIDAD & TO
BAGO,CARACAS,VENEZUELA,LA GUAYRA,VEN
EZUELA
452 DATA MARACAIBO,VENEZUELA,GEORGET
OWN,GUYANA,PARAMARIBO,SURINAM
454 DATA CAYENA,GUAYANA FRANCESA,BOG
OTA,COLOMBIA,CARTAGENA DE INDIAS,COL
OMBIA
456 DATAMEDELLIN,COLOMBIA,BRASILIA,B
RASIL,SAO PAOLO,BRASIL,RIO DE JANEIR
O,BRASIL
458 DATA QUITO,ECUADOR,GUAYAQUIL,ECU
ADOR,LIMA,PERU,CALLAO,PERU,CUZCO,PERU
460 DATA ARQUEPIA,PERU,LA PAZ,BOLIVI
A,SUCRE,BOLIVIA,COCHABAMBA,BOLIVIA
462 DATA POTOSI,BOLIVIA,ASUNCION,PAR
AGUAY,MONTEVIDEO,URUGUAY,SANTIAGO,CHILE
464 DATAVALPARAISO,CHILE,VALDIVIA,CH
ILE,BUENOS AIRES,ARGENTINA,ROSARIO,A
RGENTINA
466 DATA MENDOZA,ARGENTINA,CORRIENTE
S,ARGENTINA,HAMILTON,BERMUDAS
468 DATA WASHINGTON D.C.,ESTADOS UNI
DOS,ALASKA,ESTADOS UNIDOS
470 DATA BALTIMORE,ESTADOS UNIDOS,AL
BUNQUERQUE,ESTADOS UNIDOS
472 DATA SAN DIEGO,ESTADOS UNIDOS,SA
CRAMENTO,ESTADOS  UNIDOS
474 DATA PHOENIX,ESTADOS UNIDOS,OTTA
WA,CANADA,ONTARIO,CANADA,TORONTO,CANADA
476 DATA QUEBEC,CANADA,MONTREAL,CANA
DA,VANCOUVER,CANADA
478 DATA GUATEMALA,GUATEMALA,MONTERR
EY,MEXICO,VERACRUZ,MEXICO
```



FIRST S.A.

C/ Aribau, 62. BARCELONA- 08011
Tel. (93) 323 03 90
Tlx. 53947 FIRS E (ESPAÑA)

Ordenadores más Personales!



SE COMPLACE EN ANUNCIAR LA DISTRIBUCION DE PRODUCTOS

commodore

DE: microelectrónica y control, s. a. MEE

IMPORTADOR EXCLUSIVO PARA ESPAÑA



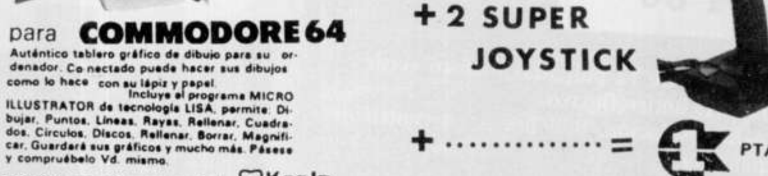


ErgoVisión
(Base móvil)
29.800 ptas.



22.105 PTAS.

KoalaPad Touch Tablet (disco)



para **COMMODORE 64**

Auténtico tablero gráfico de dibujo para su ordenador. Co nectado puede hacer sus dibujos como lo hace con su lápiz y papel.
Incluye el programa MICRO ILLUSTRATOR de tecnología LISA, permite: Dibujar, Puntos, Lineas, Rayas, Rellenar, Cuadrados, Circulos, Discos, Rellenar, Borrar, Magnificar, Guardará sus gráficos y mucho más. Pásese y compruébelo Vd. mismo.



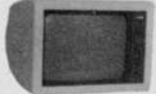


FIRST S.A. importador para españa de Koala

COMMODORE 64

+ 2 SUPER JOYSTICK

+ .... = PTAS.



METAMORPHIC
25.000 ptas.





FIRST VENDEMOS IMPRESORAS MAS PERSONALES

star

GEMINI 10 X... 79.500 Ptas.
GEMINI 15 X... 115.000 Ptas.
INCLUYE LA INTERFACE MICROANGELO Y CONECTORES (OFERTA SOLO PARA ESTE MES)
LLAME PARA SOLICITAR SU PRECIO SI NO PRECISA INTERFACE
LLAME PARA SOLICITAR SU PRECIO PARA EL RESTO DE IMPRESORAS DE LA GAMA STAR. TAMBIEN INCLUYEN LA TARJETA MICROANGELO

MICROANGELO

Es la más PODEROSA interface paralelo para su impresora Epson Star, etc. existente en el mercado mundial.

```
1 C=55304:R=56333:POKE56576,
  148:POKE53265,59:POKE53272,
  61:POKER,127:POKE1,51
2 A=52224:B=57344:FORI=0TO15
  :X=A+40×I:Y=B+320×I:FOR J=
  0TO15:POKE X+J,16×J+I:NEXT
3 K=0:FOR J=0TO127:POKE Y+J,
  PEEK (C+K):K=K+1:NEXT:NEXT
  :POKE 1,55:POKE R,129
```

CHR$(65);
CHR$(82);
CHR$(70)



Datalife
10 DISKETTES
+
2 DISKETTES
(LIMPIA CABEZAL)

LAS TARJETAS DE FIRST

```
× TARJETA VIC 20 -40/80 COL.  LLAME
× TARJETA VIC 20 -MEMO.64 K.  LLAME
× SLOTS EXPANSION V20 Y C64   LLAME
× TARJETA COMMO.64 80 COL.    LLAME
  Más de 200 programas para su
     VIC 20 y COMMODORE 64:
- Utilidades          - Lenguajes
- Sis. Operativos     - Aplicaciones
- Juegos (nacionales e importación)
( Amplio surtido de programas edu-
  cativos para sus hijos )

× CASSETE COMMODORE           LLAME
× INTERFACE RS 232            LLAME
× PADDLES                     LLAME
× FUNDA COMMODORE 64          LLAME
```

SIEMPRE NUEVAS IDEAS EN FIRST NEWS. Y LO RECIBO GRATUITAMENTE. INCREIBLE.



FABULOSO REGALO TIP POSTER

A todos los propietarios de un VIC 20 ó C 64, manden un sobre indicando: Tipo de Ordenador, anualidad y lugar donde lo compró, tipo de configuración de su equipo. Recibirán: un FABULOSO TIP POSTER COMMODORE de FIRST. Manden su carta a FIRST TIP POSTER. Y por favor, dennos tiempo a contestar todas sus cartas.



CINTAS

CINTA CASSETTE INFORMATICA F20 (1)	135 Pts
CINTA CASSETTE INFORMATICA F20 (5)	600 Pts
CINTA CASSETTE INFORMATICA F20 (10)	1.100 Pts
CINTA IMPRESORA EPSON 80	800 Pts
CINTA IMPRESORA STAR	180 Pts



```
1 INPUT"BO";A:POKE53280,A:INPUT"BA";B:POKE53281,B:INPUT
  "T";C:POKE646,C:GOTO1
```

SUPER JOYSTICK

COMPATIBLE CON COMMODORE 64 y VIC 20
SINCLAIR (precisa interf)
ATARI
SPECTRAVIDEO.

2.200!!

LOS LIBROS DE FIRST

COMMODORE-64	
Curso de Introducción al BASIC parte I	3.000
Curso de Introducción al BASIC parte II	3.000
Guia de Referencia del Programador	3.000
COMMODORE 64 COMPUTING	1.900
VIC-20	
VIC 20 GUIA DEL USUARIO	2.300
LEARN TO USE THE VIC-20	1.600
VIC-20 EXPOSED	1.900

COMPRE SIEMPRE EN FIRST

PAPEL ESPECIAL PARA VD.

2.500 Hojas de papel (bordes perfectos). Superblanco. Grueso. 80c	4.458 Ptas.
2.500 Recibos negociables standard	4.683 Ptas.
1.000 Etiquetas autoadhesivas 2 × lineas)	828 Ptas.

Todos los envios de papel son a portes debidos.

Magic Desk

Ud y MAGIC DESK, penetrarán en la nueva generación de software. Imaginese usando su ordenador para hacer archivos y editar letras personales, etc. sin aprender un simple comando. Los comandos son GRAFICOS. Mueva sólo la mano animada y realize su deseo, listo para funcionar.
MAGIC DESK en FIRST............. LLAME

First In Quality Software




ENTREGA INMEDIATA A PROVINCIAS

VENTA POR CORREO

Mande su pedido. Pago: Talón conformado o giro postal. Pedidos inferiores a 4.500 Pts. sume 180 Pts. gastos envio. Pedidos de libros sólo no sume ningun gasto de envio. Catálogo mande 200 Pts en sellos. Pedidos OFERTA DEL MES sume 180 Pts. gastos envio (salvo que pida otros articulos que no sean de oferta).
La presente lista de precios es susceptible de ser modificada sin aviso previo.

COMPARTIENDO EXPERIENCIAS ENTRE AMIGOS



Esta sección está dedicada a la colaboración de todos nuestros lectores y está dividida en dos partes:

1) Programación: Programas y similares.

2) Magia: Trucos, sugerencias, etcétera (pág. 54).

Habrá premios y alicientes para todos los participantes (ver pág. 6).

Enviarnos vuestra dirección para que podáis poneros en contacto unos con otros. ¡Animaros, chicos (...y chicas)!

Borrar

C-64

Angel Redondo García
Lago Constanza, 46-2º A
28017 MADRID

Os envío una rutina que sirve para limpiar el área de alta resolución del CBM-64 de una forma más rápida, que con el bucle FOR-NEXT que viene en el Manual de Referencia.

Se basa en utilizar la rutina Move Memory del Basic. Para ello, se hace un POKE a la última posición del área de pantalla con el valor que deseemos (0 en el ejemplo). A continuación, se utiliza la rutina Move Memory usando como dirección fuente la última posición de la pantalla de alta resolución y como dirección de destino la misma dirección menos 1. Con ello conseguimos que el valor introducido en el POKE se expanda por toda la pantalla. Después se repite el proceso con el área de video (pos. 1024-2023) que controlen el color de la pantalla en el modo BIT MAPPED.

La razón de utilizar la última posición de la pantalla y la anterior como direcciones fuente y de destino, es que la rutina Move Memory transfiere los octetos desde las posiciones altas del área seleccionada hacia las posiciones bajas de la misma, es decir, de arriba a abajo.

Como explicaba J.R.L. en el nº 6 de la revista, antes de llamar a la rutina Move Memory hay que cargar los punteros adecuados con las direcciones fuente y de destino del área a mover, que son:

DIRECCION 90-91 ($5A-$5B) Puntero de dirección fuente.

DIRECCION 88-89 ($58-$59) Puntero de dirección de destino.

Además hay que indicarle a la rutina en qué dirección debe terminar de transferir octetos. Esta dirección es, debido a la forma de funcionamiento de la rutina Move, anterior a la dirección fuente:

DIRECCION 95-96 ($5F-$60) Puntero de dirección final a mover, y no el número de octetos a transferir como se decía en el nº 6.

Por otra parte, he intentado cargar dichos punteros en Basic, mediante POKE, y me ha sido imposible, pues las posiciones 95-96, después de efectuar sobre ellas un POKE con cualquier valor, vuelven a tomar el valor 0. Probablemente se deba a que alguna rutina del BASIC utiliza dichos punteros, pero es algo que me gustaría que alguien me aclarara. Por esta razón, la carga de dichos punteros la he realizado en una pequeña rutina en código máquina, que cargo en el área del buffer del cassette mediante la siguiente rutina Basic.

```
10 POKE 53272,PEEK(53272) OR 8
20 POKE 53265,PEEK(53265) OR 32
30 POKE 16191,0:POKE 2023,3
50 READ A; IF A = -1 THEN 70
60 POKE 828 + Z, A: Z = Z + 1: GOTO 50
70 SYS828
90 DATA 169, 64, 133, 90, 169, 63, 133, 91, 169, 63, 133, 88,
   169, 63, 133, 89, 169, 1, 133, 95
100 DATA 169, 32, 133, 96, 32, 184, 163
110 DATA 169, 232, 133, 90, 169, 7, 133, 91, 169, 231, 133, 88,
   169, 7, 133, 89, 169, 1, 133, 95
120 DATA 169, 4, 133, 96, 32, 184, 163, 96, -1
130 GOTO 130
```

Descripción:

Linea

10-20 Se activa el modo alta resolución.

30 Se ponen los valores deseados en las posiciones altas del área de alta resolución (pos. 16191) y el área de video (pos. 2023) que controla el color de la pantalla.

50-60 Carga la rutina en C.M. en el buffer del cassette.

70 Se llama a la subrutina.

90-120 Rutina en C.M.

130 Mete a la rutina en un loop del que se sale pulsando RUN/STOP y RESTORE. Esta instrucción de eliminarse cuando se incluye la rutina en un programa completo o sustituirla por un RETURN para utilizar como subrutina.

La rutina en ensamblador es la siguiente:

```
LDA #$40   Cargamos el puntero de dirección fuente
STA  $5A   (90-91=$5A-5B) con la dirección final del área de
LDA #$3F   alta resolución + 1(16192=$3F 40)
STA  $5B

LDA #$3F   Cargamos el puntero de dirección de destino
STA  $58   (88-89=$58-$59) con la dirección anterior del final
LDA #$3F   del área de alta resolución + 1(16190+1=$3F3F)
STA  $59
```

LDA	#$01	Cargamos el puntero de dirección final a transferir
STA	$5F	(95-96=$5F-$60) con la dirección del principio
LDA	#$32	del área de alta resolución + 1 (8192+1=$3201)
STA	$60	
JSR	$A3B8	Llamamos a la rutina Move Memory
LDA	#$E8	Cargamos el puntero de dirección fuente con la
STA	$5A	dirección final del área de pantalla + 1
LDA	#$07	(2023 + 1 = $07E8)
STA	$5B	
LDA	#$E7	Cargamos el puntero de dirección de destino con
STA	$5B	la dirección anterior al final del área de pantalla
LDA	#$07	+ 1(2022 + 1 = $07E7)
STA	$5B	
LDA	#$01	Cargamos el puntero de dirección final a transferir
STA	$5F	con la dirección del principio del área de pantalla
LDA	#$04	+ 1(1024 + 1 = $0401)
STA	$60	
JSR	$A3B8	Llamada a la rutina Move Memory
RTS		Volvemos a la rutina Basic

Podéis comparar el tiempo de ejecución de esta rutina con la que viene en el Manual de Referencia en Basic y veréis que merece la pena intentarlo en lenguaje máquina.

```
10 POKE 53272,PEEK(53272)OR8
20 POKE 53265,PEEK(53265)OR 32
30 POKE 16191,0:POKE 2023,3
50 READ A:IF A=-1 THEN 70
60 POKE 828+Z,A:Z=Z+1:GOTO 50
70 SYS828
90 DATA 169,64,133,90,169,63,133,91,
169,63,133,88,169,63,133,89,169,1,13
3,95
100 DATA 169,32,133,96,32,184,163
110 DATA 169,232,133,90,169,7,133,91
,169,231,133,88,169,7,133,89,169,1,1
33,95
120 DATA 169,4,133,96,32,184,163,96,
-1
130 GOTO 130
```



Voyager

VIC-20
CON O SIN SUPEREXPANDER

Carlos Rojas López
Gustavo Gallardo, 7-4º D
Telf.: (954) 61 41 24
Sevilla

Mando este programa de juegos, bastante sencillo. Como estáis hartos de piropos no os mando ninguno.

El juego es para el VIC 20 no ampliado, pero sin embargo en la primera parte del programa aparecen sentencias del SUPER EXPANDER; por consiguiente no podrá ser utilizado por el VIC standar, a no ser que se suprima este primer programa y con él, la línea 30 del segundo programa.

El juego es bastante simple. Consiste, como su nombre indica (VOYAGER), en un viaje inter-espacial atravesando la galaxia y comiendo marcianitos que te encuentres a tu paso. Deberás tener cuidado con los terribles bloques de asteroides que te destruirán. Cuando te hayas comido un número de marcianitos, incrementados al pasar de fases, pasarás a otra pantalla más peligrosa, con menos marcianitos y con más bloques. Cada diez puntos, nave extra. El juego tiene 10 fases y a la onceava, tendrás que aterrizar en una pequeña base en donde se acabará el juego. Las teclas de movimiento son:

—Arriba f1
—Abajo f7

Sin más, paso a describiros el programa que consta de dos partes. (Primero deberás teclear la primera parte y salvarla en el casset, luego copiar la segunda parte y salvarla también. Rebobinar la cinta al principio y pulsar SHIFT y RUN/STOP).

1ª parte:

10-10: Comentario: nombre del programa.
30: Modo gráfico 2 del Super Expander. Color de pantalla, fondo, auxiliar y carácter.
40: Draw que forma la V.
50: Circle que forma la O.
60: Draw que forma la Y y la A.
70: Draw que forma, junto con Circle, forman la G.

GLOSARIO

Analista.—Una persona que analiza y define los problemas, y desarrolla algoritmos y procedimientos para su solución.

AND.—Un operador lógico que hace que siendo X una premisa, Y otra premisa, y Z otra, el resultado de AND de Y, X, Z será cierto si todas las premisas lo son, y falso si cualquiera de ellas lo es.

APL (A Programming Language).—Es un lenguaje de programación con un juego de caracteres y una sintaxis muy peculiar, fue diseñado principalmente para aplicaciones matemáticas o para cuando se necesita manejar gran cantidad de matrices numéricas o de caracteres.

Append.—Es uno de los comandos de la unidad de disco. Añadir un archivo o elemento al final de otro.

Argumento.—Variable o parámetro independiente que es transferido a otro programa o rutina.

ASCII.—American National Standard Code for Information Interchange. Es el código estándar utilizado para el intercambio de información entre distintos dispositivos o equipos, utiliza 7 bits para codificar un caracter (8 si incluimos el de paridad), y su juego de caracteres esta compuesto de caracteres de control y alfanuméricos.

Backup.—Relativo a un sistema, dispositivo, archivo o posibilidad de obtener una copia de seguridad para evitar que por error o fallo de funcionamiento se pierdan datos.

BAM.—Mapa de disponibilidad de bloques, lo utiliza la unidad de disco para saber qué bloques están ocupados o libres en un disco.

BCD.—Notación decimal codificado en binario.

BIN.—Binario. Relativo a una selección con dos posibles estados. Sistema de numeración con base dos.

80: Draw que forma parte de la E.
90: Draw que forma parte de la E y la R.
100: Draw que forma la otra parte de la R.
110-150: Rutinas de presentación.
160: Carga (LOAD) automática.

2ª parte:

10: Rótulo.
20: Borra la pantalla gráfica del Super Expander y vuelve al modo de texto normal del VIC.
30: Protección del juego de caracteres.
40: Colocación de los pixels de la forma 8×8, para los caracteres de 0 a 15: de la Arroba@ hasta la O, según el juego 1 del código POKE.
60-220: Datas de creación de caracteres programables.
220: Presentación del juego.
230: Inicia al VIC al modo de caracteres programables.
250: Barrido de basura de la pantalla gráfica.
260-265: Variables.
270: Colocación de bordes laterales.
280-320: Marcianitos de puntuación.
330-370: Imprime obstáculos: bloques.
390: Indica naves disponibles y puntuación.
400: Indicador de fases.
410: Tecla pulsada.
450: Cada cinco puntos cambia de fase.
490: Retorno de bucle.
500-570: Subrutina que coloca barreras laterales.
600-690: Subrutina de tecla pulsada.
610: hace avanzar la nave hacia adelante. Si llega al final de la pantalla, vuelve al principio.
620-640: Asciende o desciende la nave.
650: Comprueba si encuentras un marcianito. Entonces a li.800.
660: Si choca contra un bloque, va a la línea 750.
670: Si en la fase once, consigues aterrizar en la base, va a la línea 1500.
680: Coordenadas de movimiento de la nave.
690: Retorna al programa principal.
750-780: Colisión contra un bloque:
760: Sonido de explosión de la nave.
770: Controla el número de naves auxiliares. Si son cero, acaba el juego.
780: Retorno al bucle principal.
800-840: Puntuación al comer marcianitos.
810: Sonido de destrucción de los marcianos.
820-830: Aumentan los puntos. Cada diez puntos nave extra.
900-930: Cambio de fase.
905: Indica que pasa a la fase de aterrizaje. Si Z1=0.
910: Borra bloques y marcianitos para pasar a otra fase.
920: Reconversión de variables:
Aumentan bloques.
Disminuyen marcianitos.
Si aún no es la última fase, indica que aparezcan nuevos marcianitos y bloques.
930: En la última fase, pinta la pista de aterrizaje.
1000-1060: Fase de aterrizaje.
1010-1030: Dibuja barrera de fondo y pista, sólo la primera vez. Si DD=0.
1040: Impide que se vuelva a pintar lo antes dicho.
1060: Vuelta al programa conductor.
1500-1560: Llegada después de aterrizar.
1510: Dibuja la nave ya aterrizada.
1520: Mensaje de felicitación.
1530: Introduces el nombre.
2000-2150: Subrutina de presentación.
2300-2350: Subrutina de sonido de nave extra.
2500-2510: Final de juego.

Primera parte

```
10 REM VOYAGER,1.PARTE
20 REM ROTULO
30 GRAPHIC2:COLOR2,2,7,1
40 DRAW2,20,50TO70,400:DRAW2TO120,50

50 CIRCLE2,220,220,50,180:DRAW2,320,
50TO370,200:DRAW2TO420,50
60 DRAW2,362,200TO362,400:DRAW2,450,
400TO510,50:DRAW2TO560,400:DRAW2,480
,250TO535,250
70 CIRCLE2,640,220,50,180,5,88:DRAW2
,685,265TO640,265:DRAW2,850,50TO740,
50
80 DRAW2TO740,400:DRAW2TO850,400:DRA
W2,750,225TO800,225
90 DRAW2,890,200TO990,200:DRAW2TO990
,50:DRAW2TO890,50:DRAW2TO890,400
100 DRAW2,890,200TO990,400
110 REGION1:CHAR10,6,"[SHIFTO][7COMMT]
[SHIFTP]":CHAR11,6,"[COMMG][2SPC]IS[3SPC]
[COMMM]"
120 CHAR12,6,"[COMMG]LOADING[COMMM]"

130 CHAR13,6,"[SHIFTL][7COMM@][SHIFT@]
"
140 CHAR16,1,"NEW[SPC]GENERATION[SPC]
OF"
150 CHAR18,0,"SOFWARE[SPC](C)S.A.,19
84"
160 POKE631,131:POKE198,1
```

Segunda parte

```
10 REM VOYAGER
20 SCNCLR:GRAPHIC0
30 POKE52,28:POKE56,28:CLR
40 FORT1=7168TO7168+8*16-1:READA1:PO
KET1,A1:NEXT
60 DATA0,224,156,130,137,130,156,224

70 DATA126,255,219,255,255,195,255,1
26
80 DATA129,66,36,24,126,255,165,36
90 DATA24,24,126,189,60,36,36,102
100 DATA 60,126,219,255,255,255,255,
165
110 DATA0,102,153,24,126,153,36,66
120 DATA60,126,219,255,255,255,255,1
65
130 DATA0,126,129,189,165,189,129,12
6
140 DATA0,9,25,127,255,24,8,0
150 DATA129,153,189,255,255,189,153,
129
160 DATA165,231,66,255,219,255,255,9
0
170 DATA126,255,255,255,255,255,255,
126
180 DATA0,0,0,0,0,0,0,255
190 DATA255,1,1,1,1,1,1,255
200 DATA129,129,129,129,129,129,129,
129
210 DATA0,0,0,0,0,0,0,0
220 GOSUB2000
230 POKE36869,255
250 FORQW=0TO505:POKE7680+QW,15:NEXT

260 POKE36879,110:C1=5:B1=10:D1=7746
:E1=38466:F1=8098:G1=38818:M1=23:Q1=
1:JJ=7753
265 KK=38473:Z1=1:P1=0
270 GOSUB500:X1=0:Y1=8
280 REM MARCIANITOS
290 FORK1=1TOM1
300 H1=INT(RND(1)*14):I1=INT(RND(1)*
17)
320 POKEJJ+H1+I1*22,Q1:POKEKK+H1+I1*
22,4:NEXT
330 REM BOLQUES
340 FORV1=1TOB1
350 L1=INT(RND(1)*14):N1=INT(RND(1)*
17)
360 IFPEEK(JJ+L1+N1*22)=Q1THEN350
370 POKEJJ+L1+N1*22,11:POKEKK+L1+N1*
22,7:NEXT
390 PRINT"[HOM][RVSON][YEL]CRAFTS:";
C1"[CRSRR]POINTS:"P1
400 PRINT"[HOM][21CRSRD][2CRSRR][RVSON]
[GRN]FASE";Q1
410 GOTO600
```

Nuevo curso de ordenador personal y programación Basic

SI VAS A SEGUIR VIVIENDO EN ESTE PLANETA NO TE QUEDES DESCONECTADO

¿Cómo dominar a los ordenadores antes de que ellos dominen todos los sectores de la vida?

Si dominas tu impaciencia, ya tienes medio curso hecho



Nuevo curso a distancia Ordenador Personal y Programación Basic

Ahora puedes hacerlo de una forma sencilla y divertida. Con un dedo de la mano. Sin saber una palabra de inglés. Y sin moverte de casa.

Gracias a un curso claro y entretenido que te ayuda a sacarle al ordenador todo lo que tiene dentro.

Por supuesto que no te estamos hablando de un Manual de Instrucciones, y allá tú. Se trata de un programa de aprendizaje muy práctico, diseñado por especialistas, para que nadie se aburra a mitad de camino.

Es, para que te hagas una idea, como un viaje organizado al futuro. ¿Quieres una plaza?

Te ofrecemos 12 manuales de lecciones explicadas paso a paso.

Te ofrecemos, si no tienes ordenador, el COMMODORE 64, el más interesante del mundo, para que practiques desde el primer momento. Te ofrecemos un cuadro de profesores que te orientan continuamente en tu aprendizaje. ¿Qué pones tú? Paciencia. Y un poco de voluntad.

Con esos dos instrumentos te garantizamos que en seis meses serás capaz de operar con cualquier programa que adquieras y algo que te gustará más: podrás crear tus propios programas para uso profesional o personal: gestión, administración, archivo, estadística, juegos, estudios... Verás qué divertido es tener paciencia.

Este es un curso diferente a todos los que conozcas. Es serio, pero sabiendo que no eres una máquina. Es divertido, pero sin olvidar que tienes que sacarle partido.

Consta de los siguientes elementos:

- 12 manuales de lecciones, preparados para aprender enseguida sin perderse en teorías.
- 1 anexo de equivalencias.
- Optativo: Un ordenador personal COMMODORE 64, con su unidad de casettes para practicar desde el primer minuto, porque como se aprende a dominar los ordenadores es trabajando con ellos.
- Amplio número de programas y una orientación continuada de tus profesores.
- No son necesarios conocimientos especiales de ningún tipo.
- Duración = Seis meses aproximadamente dedicando dos horas diarias.
- Sistema de aprendizaje: A distancia.

Cada manual incluye ejercicios amenos para realizar en el ordenador. El profesor contesta a vuelta de correo con las correcciones y los consejos necesarios.

- Precio: El curso completo por algo más del precio del propio ordenador. Y con facilidades de pago.



Centro de Enseñanza a Distancia autorizado en aplicación del Decreto 2641/1980.

Otros de los cursos CCC:

- Electrónica.
- Radio, TV.
- Técnico en Energía Solar.
- Instalador Electricista.
- Técnico en Fontanería.
- Graduado Escolar.
- Guitarra.
- Contabilidad.
- Mecánico de motos.
- Dibujante de comics.
- Inglés (con casette o por video).
- Auxiliar Enfermera.
- Puericultura.
- Estheticiene.
- Peluquería.
- Masaje y Digitopresión.
- Corte y Confección.

Para hablar con los dedos, te echamos una mano

El idioma del futuro, se practica con los dedos. Si quieres aprenderlo, la oportunidad está en tu mano: Envía este cupón.

Deseo información **gratis y sin compromiso** sobre el curso de:

____

Nombre y apellidos ______________________

____

Dirección ______________________

______________ N.º ______ Piso ______

Población ______________________

Provincia ______________ Cod. Post. ______

Teléfono ______________ Edad ______

Envía este cupón a:

CCC, alto de Miraconcha - Apdo. 666 - 20080
Tfno.: (943) 467600 - SAN SEBASTIAN
o a **CCC**, Apdo. 17.222 - 28080-MADRID

C6-41-83-48-84

```
450 IFP1>=5+NNTHEN900
490 GOTO390
500 REM BARRERAS LATERALES
510 FORQQ=0TO21:POKE7724+QQ,11:POKE3
8444+QQ,0:NEXT
520 POKED1,11:POKED1+1,11:POKED1+20,
11:POKED1+21,11:POKED1+22,11:POKED1+
43,11
530 POKEE1,0:POKEE1+1,0:POKEE1+20,0:
POKEE1+21,0:POKEE1+22,0:POKEE1+43,0
540 POKEF1-22,11:POKEF1-1,11:POKEF1,
11:POKEF1+1,11:POKEF1+20,11:POKEF1+2
1,11
550 POKEG1-22,0:POKEG1-1,0:POKEG1,0:
POKEG1+1,0:POKEG1+20,0:POKEG1+21,0
560 FORQW=0TO21:POKE8120+QW,11:POKE3
8840+QW,0:NEXT
570 RETURN
600 REM TECLA PULSADA
610 X1=X1+1:IFX1>21THENB1=8:X1=0:GOT
O330
620 O1=PEEK(197)
630 IFO1=39THENY1=Y1-1
640 IFO1=63THENY1=Y1+1
650 IFPEEK(7746+X1+Y1*22)=Q1THENGOSU
B800
660 IFPEEK(7746+X1+Y1*22)=11THEN750
670 IFPEEK(7746+X1+Y1*22)=13THEN1500
680 POKE7746+X1+Y1*22,0:POKE38466+X1
+Y1*22,7:POKE7746+X1+Y1*22,15
690 GOTO450
750 REM CHOQUE
760 POKE36878,15:POKE36877,210:FORTT
=1TO300:NEXT:POKE36877,0:FORT=1TO500
:NEXT
770 C1=C1-1:IFC1=0THEN2500
780 X1=0:Y1=8:GOTO390
800 REM PUNTUACION
810 POKE36878,15:POKE36876,240:FORTT
=1TO100:NEXT:POKE36876,0
820 P1=P1+1
830 IFP1>=10+GGTHENC1=C1+1:GG=GG+10:
GOSUB2300
840 RETURN
900 REM CAMBIO DE FASE
905 IFQ1=10THENZ1=0
910 FORCA=0TO374:POKE7753+CA,15:NEXT
:GOSUB520:NN=NN+5+Q1:M1=M1-1:Q1=Q1+1
920 X1=0:Y1=8:B1=10:B1=B1+Q1+4:IFZ1=
1THEN280
930 GOTO1000:GOTO390
1000 REM ATERRIZAJE
1005 JJ=8098+7:KK=38818+7
1010 IFDD=0THENFORFF=1TO13:POKE7789+
FF*22,11:POKE38509+FF*22,0:NEXT
1020 IFDD=0THENPOKEJJ,11:POKEJJ-21,1
1:POKEJJ-22,11:POKEJJ-23,11:POKEJJ-4
3,13
```



Simon's Grafic

C-64

Luis Pellicer Lorenzo
Conde de Torrecedeira, 122-3º A
36108 VIGO (Pontevedra)

Son programas realizados con el cartucho de ampliación de Basic, el SIMON'S BASIC.

Lo que más echamos de menos algunos de los usuarios del 64 son los comandos que permitan trabajar en alta resolución de forma sencilla y que el SIMON'S incorpora. Por eso como pequeño ejemplo de sus capacidades, mando estos dos primeros y pequeños programas grabados con dicho cartucho.

El primero dibuja una función trigonométrica dándole dos parámetros y el segundo es una pequeña demostración de la posibilidad de crear bonitos dibujos en alta resolución con una facilidad de programación asombrosa.

Los que lo hemos intentado sabemos el montón de líneas en BASIC que serían necesarias para hacer esto mismo y la lentitud con que serían ejecutados. Ambos problemas son sobradamente obviados con el SIMON'S.

Y esto es una de las muchas facetas en donde el SIMON'S llena el hueco que en su configuración mínima posee.

```
100 AB=0:COLOUR0,0:PRINT"[CLR][YEL]
"
105 PRINTAT(13,8)"REPRESENTACION":P
RINT
110 CENTRE"GRAFICA[SPC]DE[SPC]FUNCI
ONES":PRINT:PRINT
115 CENTRE"TRIGONOMETRICAS"
120 N=10:EXEC ESPERA
125 PRINT"[CLR]"
130 PRINTAT(5,8)"INTRODUCE[SPC]PARA
METROS"
135 PRINTAT(8,10)"A[SPC]:[SPC]";
140 FETCH"[CRSRD]",3,A
145 PRINTAT(8,12)"B[SPC]:[SPC]";
150 FETCH"[CRSRD]",3,B
155 PRINTAT(10,17)"SON[SPC]CORRECTO
S[SPC](S/N)[SPC]";
160 FETCH"[CRSRR]",1,A$
165 IFA$="N"THEN125
170 IFA$<>"S"THEN155
175 PRINT"[CLR]"
180 Q=100:SQ=160:SW=100:W=3:Z=1.25
185 IFAB=0THENHIRES1,0
190 CSET2
195 FORT=0TO2*πSTEPπ/180
200 M=Q*SIN(T*W)
205 X=SQ-INT(M*Z*COS(A*T))
210 Y=SW-INT(M*SIN(B*T))
215 PLOTX,Y,1
220 NEXT
225 TEXT8,1,"[CTRL A]P[CTRL B]ULSA[SPC]
[CTRL A]RETURN[CTRL B][SPC]O[SPC]ESP
ERA",1,1,7
230 PAUSE60
235 TEXT8,1,"[CTRL A]P[CTRL B]ULSA[SPC]
[CTRL A]RETURN[CTRL B][SPC]O[SPC]ESP
ERA",0,1,7
240 NRM
245 PRINTAT(4,9)"OTRO[SPC]DIBUJO[SPC]
(S/N)[SPC]";
250 FETCH"[CRSRR]",1,A$
255 IFA$<>"S"ANDA$<>"N"THEN245
260 IFA$="S"THEN270
265 PRINT"[CLR]":END
```

```
1030 IFDD=0THENPOKEKK,2:POKEKK-21,2:
POKEKK-22,2:POKEKK-23,2:POKEKK-43,7
1040 DD=1
1060 GOTO390
1500 REM LLEGADA
1510 POKEJJ-43,0:POKEKK-43,7
1520 PRINT"[HOM][9CRSRD][5CRSRR][RVSON]
WELL[SPC]DONE[SPC]!!":FORT=1TO2000:N
EXT
1530 POKE36869,240:N$="":INPUT"[CLR].
[3CRSRD][CRSRR][RVSON]ENTER[SPC]YOUR
[SPC]NAME:";N$
1560 FORT=1TO7000:NEXT:GOTO230
2000 REM PRESENTACION
2010 POKE36879,25:PRINT"[CLR][RVSON]
[GRN][6CRSRR]VOYAGER"
2015 POKE646,32
2020 PRINT"[2CRSRD][CRSRR]DEBES[SPC]
COMERTE[SPC]EL[SPC]N."
2030 PRINT"[CRSRR]MAYOR[SPC]DE[SPC]M
ARCIANITOS"
2040 PRINT"[CRSRR]Y[SPC]EVITAR[SPC]C
OLISIONAR"
2050 PRINT"[CRSRR]CONTRA[SPC]LOS[SPC]
BLOQUES."
2060 PRINT"[CRSRD][CRSRR]CADA[SPC]5[SPC]
MUNECOS[SPC]COMI"
2070 PRINT"[CRSRR]DOS[SPC]PASARAS[SPC]
A[SPC]OTRA"
2080 PRINT"[CRSRR]FASE[SPC]Y[SPC]CAD
A[SPC]10[SPC]PUN-"
2085 PRINT"[CRSRR]TOS[SPC]NAVE[SPC]E
XTRA."
2090 PRINT"[CRSRD][CRSRR]EL[SPC]JUEG
O[SPC]CONSTA[SPC]DE[SPC]3"
2100 PRINT"[CRSRR]FASES[SPC]Y[SPC]EN
[SPC]LA[SPC]ULTIMA"
2110 PRINT"[CRSRR]TENDRAS[SPC]QUE[SPC]
ATERRIZAR"
2120 PRINT"[CRSRR]EN[SPC]UNA[SPC]PEQ
UENA[SPC]BASE."
2130 PRINT"[2CRSRD][6CRSRR][RVSON]PU
LSA[SPC][RED]←[BLU]"
2140 WAIT197,31
2150 PRINT"[CLR]":RETURN
2300 REM SONIDO NAVE EXTRA
2310 POKE36878,15
2320 FORSO=1TO2:POKE36875,240:FORT=1
TO50:NEXT:POKE36875,0:FORT=1TO150:NE
XT
2330 POKE36875,240:FORT=1TO50:NEXT:P
OKE36875,0:FORT=1TO300:NEXT
2340 NEXTSO
2350 RETURN
2500 REM FINAL JUEGO
2510 PRINT"[HOM][10CRSRD][RVSON][6CRSRR]
GAME[SPC]OVER"
2520 FORT=1TO7000:NEXT:GOTO230
```

```
270 PRINTAT(4,11)"ENCIMA[SPC]DEL[SPC]
ANTERIOR[SPC](S/N)[SPC]";
275 FETCH"[CRSRR]",1,B$
280 IFB$="S"THENAB=1:GOTO125
285 IFB$="N"THENAB=0:GOTO125
290 GOTO270
295 END
300 PROC ESPERA
305 PAUSE "[HOM][22CRSRD][5CRSRR][RVSON]
[5SPC]PULSA[SPC]RETURN[SPC]O[SPC]ESP
ERA[5SPC][RVSOFF]",N
310 END PROC
```

```
100 COLOUR0,0
105 HIRES1,0
110 FORT=0TO200
115 LINET,T,200-T,200,2
120 NEXT
125 FORT=0TO200
130 LINE200-T,200,T,200-T,2
135 NEXT
140 FORT=0TO100
145 LINE100-T,100-T,100,T,2
150 LINE100+T,100-T,100,T,2
155 NEXT
160 FORT=0TO100
165 LINE100-T,100-T,T,0,2
170 LINE100+T,100-T,200-T,0,2
175 NEXT
180 BLOCK0,0,200,200,2
185 BLOCK0,0,200,200,2
190 BLOCK0,0,200,200,2
195 BLOCK0,0,200,200,2
200 PAUSE1000

READY.
```



Battle Zone

VIC-20 + 3K

José Ignacio Aranda Hevia (15 años)

"BATTLE ZONE", es una lucha en el desierto en la que debes matar a los tanques enemigos que se multiplican al avanzar el grado de dificultad.

Tu tanque se mueve con la palanca de juegos hacia la derecha hace girar el tanque en el sentido de las agujas del reloj, hacia la izquierda en sentido contrario, para avanzar hacia atrás, se dispara, como es normal, con el botón de disparo. Dispones de dos tanques en cada pantalla.

Se necesita una expansión de memoria de tres k.

```
1 POKE650,1:GOTO14010
10 TV=1:FORT=1TO8
11 TA$(T)=CHR$(T+68):NEXT
50 FORE=1TORF:POKE7680+C+B,(TA+5):PO
KE7680+C+B+TV,160
51 GOSUB1500
52 IFFB=1THEN6000
53 IFYI<>0THENPOKE36877,140:GOTO55
54 POKE36877,0
55 WD=13:IFRF>0THENGOSUB7000
56 GOSUB12000
57 NEXTE:GOTO50
1000 POKE52,24:POKE56,24
1001 POKE36869,255
1002 FORT=1TO136
1003 READA:POKE7167+T,A:NEXT:GOTO110
0
1050 DATA255,192,160,144,136,132,130
,129,255,3,5,9,17,33,65,129
1051 DATA129,130,132,136,144,160,192
,255
1052 DATA129,65,33,17,9,5,3,255,255,
195,165,153,153,165,195,255
1053 DATA8,8,251,235,235,235,235,255
,0,2,8,252,236,168,236,252
1054 DATA0,252,236,234,236,236,252,0
,252,236,168,236,252,8,2,0
1055 DATA255,255,235,235,235,251,8,8
,63,59,42,59,63,32,128,0
1056 DATA0,63,59,59,171,59,59,63,0,1
28,163,59,42,59,63,0
1059 DATA12,12,42,42,42,42,34,51,0,0
,195,40,40,40,195,0
1060 DATA0,0,20,60,20,0,0,0,24,36,12
6,102,102,60,24,36,102
1100 PRINT"[HOM]"
1105 FORQ=1TO3
1110 A=INT(RND(1)*18):IFA=FTHEN1110
1111 D=INT(RND(1)*3)+1:FORW=1TOD:PRI
NT:NEXTW:GOSUB1200
1115 NEXTQ:GOTO1300
1200 POKE646,6:PRINTTAB(A)"[WHT][RVSOFF]
@[RVSON][COMMT][RVSOFF]A"
1201 PRINTTAB(A)"[RVSON][COMMG][RVSON]
[SHIFTV][RVSOFF][RVSON][COMMM]"
1203 PRINTTAB(A)"[RVSOFF]B[RVSON][COMM@]
[RVSOFF]C":RETURN
1300 FORT=1TO5
1301 A=INT(RND(1)*505):IFPEEK(7680+A
)<>160THEN1301
1302 POKE7680+A,4:POKE38400+A,1
1303 NEXT:GOTO14102
1500 POKE37154,127:XT%=PEEK(37152)AN
D128:POKE37154,255
1501 XT%=XT%OR(PEEK(37137)AND127)
1502 XI=SGN(XT%AND128)-SGN(XT%AND16)
1503 YI=SGN(XT%AND8)-SGN(XT%AND4)
1504 TG=1-SGN(XT%AND32):IFTG=1THENK=
C:L=B:GOTO5000
1519 IFXI=0THENPOKE36875,0:GOTO1522
1520 TA=TA-XI:IFTA=8THENTA=1
1521 POKE36875,140:IFTA=-1THENTA=7
1522 IFYI=-1THEN2000
1525 RETURN
1800 POKE646,14:PRINT"[CLR][RVSON][21COMM+]
"
1801 FORT=1TO19
1802 PRINT"[RVSON][COMM+][19SPC][COMM+]
":NEXT
1803 PRINT"[RVSON][21COMM+]":RETURN
2000 ON(TA+1)GOTO2050,2080,2100,2150
,2180,2200,2230,2270
2050 IFPEEK(7680+C+B-22)<>160THEN140
00
2051 B=B-22:TV=22:RETURN
2080 IFPEEK(7680+C+B-21)<>160THEN140
00
2081 B=B-22:C=C+1:TV=21:RETURN
2100 IFPEEK(7680+C+B+1)<>160THEN1400
0
2101 C=C+1:TV=-1:RETURN
2150 IFPEEK(7680+C+B+1+22)<>160THEN1
4000
2151 C=C+1:B=B+22:TV=-23:RETURN
2180 IFPEEK(7680+C+B+22)<>160THEN140
00
2181 B=B+22:TV=-22:RETURN
2200 IFPEEK(7680+C+B+22-1)<>160THEN1
4000
2201 B=B+22:C=C-1:TV=-21:RETURN
2230 IFPEEK(7680+C+B-1)<>160THEN1400
0
2231 C=C-1:TV=1:RETURN
2270 IFPEEK(7680+C+B-1-22)<>160THEN1
4000
2271 C=C-1:B=B-22:TV=23:RETURN
5000 ON(TA+1)GOTO5010,5020,5030,5040
,5050,5060,5070,5080
5010 FI=-22:DI=22:FB=1:RETURN
5020 FI=-21:DI=21:FB=1:RETURN
5030 FI=+1:DI=-1:FB=1:RETURN
5040 FI=23:DI=-23:FB=1:RETURN
5050 FI=22:DI=-22:FB=1:RETURN
5060 FI=21:DI=-21:FB=1:RETURN
5070 FI=-1:DI=1:FB=1:RETURN
5080 FI=-23:DI=23:FB=1:RETURN
6000 MO=MO+FI:IFPEEK(7680+K+L+MO)<>1
60THENPOKE7680+K+L+MO-FI,160:FB=0:GO
TO9000
6001 POKE7680+K+L+MO,15:POKE7680+K+L
+MO+DI,160:GOTO53
7000 WC=0:CW=0:IFJ(E)=CTHEN7500
7001 IFJ(E)<CTHENJ(E)=J(E)+1:WC=-1:G
OTO7003
7002 J(E)=J(E)-1:WC=1
7003 IFH(E)+J(E)=C+BTHEN14000
7004 GOTO7800
7500 IFH(E)=B+22ORH(E)=B-22ANDJ(E)=C
THEN14000
7501 IFH(E)>BTHENH(E)=H(E)-22:CW=22:
```

INFORMATICA Y EDUCACION



- CIA. ESPECIALIZADA EN EQUIPOS, SOFTWARE Y SERVICIOS PARA LA ENSEÑANZA TANTO DE INFORMATICA COMO DE OTRAS MATERIAS.
- DISTRIBUYE CON CONTRATO EN EXCLUSIVA PARA TODA ESPAÑA:

☐ **Red local ED NET desarrollada por BSP,** capaz de conectar hasta 250 puestos al ordenador central, siendo éste un C-64. También permite que estos puedan utilizar dispositivos de disco duro o flexible, etc. y que desde un punto master se controlen todos y cada uno de los puestos-alumno.

☐ **Control de automatismos.** Permite, desde el C-64, controlar 32 parámetros de entrada y otros tantos de salida de la máquina o máquinas que se quieran automatizar.

☐ **Control de robots.** Permite desde el C-64 controlar robots.

☐ **Pantalla digital.** Maneja el ordenador al tacto de la pantalla.

☐ **Paquete Mágico.** El profesor puede preparar cualquier asignatura para ser enseñada y controlada a través del ordenador.

INFORMATICA Y EDUCACION

Teléfono (91) 401 29 50
C/ Gral. Pardiñas, 74
28006 Madrid

Teléfono (93) 247 40 03
C/ Joaquín Folguera, 2, bajo
Barcelona

```
GOTO7503
7502 H(E)=H(E)+22:CW=-22:GOTO7900
7503 IFH(E)+J(E)=C+BTHEN14000
7504 GOTO7900
7800 IFPEEK(7680+J(E)+H(E))<>160THEN
J(E)=J(E)+WC:WC=0:GOTO7501
7900 IFPEEK(7680+J(E)+H(E))<>160THEN
H(E)=H(E)+CW:RETURN
8000 POKE7680+H(E)+J(E),WD:POKE7680+
H(E)+J(E)+WC+CW,160:RETURN
9000 PO=K+L+MO:MO=0:GV=RF:SC=0
9010 FORHB=1TO2:FORYE=SCTOGV
9011 IFPO=H(YE)+J(YE)THENGOTO9040
9012 NEXTYE:SC=WZ+1:GV=FV+WZ+1:NEXTH
B:GOTO53
9040 POKE7680+J(YE)+H(YE),160
9041 FORGH=YETOGV:H(GH)=H(GH+1):J(GH
)=J(GH+1):NEXTGH:RF=RF-1:IFHB<>1THEN
RF=RF+1:FV=FV-1
9042 POKE36876,180:IFRF+FV=0THENREST
ORE:POKE36876,0:GOTO14010
9043 POKE36876,0:GOTO53
9100 T=0
9101 FORT=1TOINT(RF*2)
9102 DF=INT(RND(1)*17)+1
9103 FD=INT(RND(1)*17)+1:FD=FD*22
9104 IFPEEK(7680+DF+FD)=160THENH(T)=
FD:J(T)=DF:POKE7680+FD+DF,16:POKE384
00+DF+FD,3:GOTO9106
9105 GOTO9102
9106 NEXT:GOTO9150
9150 DF=INT(RND(1)*17)+1
9151 FD=INT(RND(1)*17)+1:FD=FD*22
9152 IFPEEK(7680+DF+FD)=160THENC=DF:
B=FD:RETURN
9153 GOTO9150
11000 IFRF>1THENWD=14:E=E+RF:GOSUB70
00:E=E-RF
12000 IFFV=0THENRETURN
12001 VF=VF+1:IFVF>FVTHENVF=1
12002 WD=14:FD=E:E=WZ+VF:GOSUB7000:E
=FD
12005 IFRND(1)>.5THEN12007
12006 RETURN
12007 WM=H(WZ+VF)+J(WZ+VF)+WC+CW
12008 POKE7680+WM,16:POKE38400+WM,0:
RETURN
14000 POKE7680+B+C,17:POKE38400+B+C,
6
14001 POKE36877,220
14002 PN=PN-1:IFPN<=0THEN14050
14003 FORR=15TO1STEP-1:FORT=1TO50:PO
KE36878,R:NEXTT:NEXTR
14004 POKE36877,0:POKE36878,15:HB=1
14005 POKE7680+B+C,87
14006 GOTO14100
14010 CLR:PAN=PEEK(650):POKE650,PAN+
1:DIMH(40),J(40),WZ(40):PN=2:GOTO141
00
14050 POKE36869,240:POKE646,6:PAN=PE
EK(650):POKE36878,0
14051 PRINT"[CLR]SE[SPC]ACABO[SPC]"
14052 PRINT"[CRSRD]HAS[SPC]LLEGADO[SPC]
A[SPC]LA[SPC]"
14053 PRINT"[CRSRD]PANTALLA";PAN
14054 END
14100 RESTORE:GOSUB1800:POKE646,14:G
OTO1000
14102 POKE646,14:RF=PAN:FV=PAN:WZ=PA
N
14103 POKE36878,10:POKE36879,149:POK
E36878,(PEEK(36878)AND15)OR(16*PAN):
E=0
14105 POKE36875,0:POKE36877,0:GOSUB9
100
14107 FORWH=1TO20
14108 POKE7680+C+B,(TA+5)
14109 GOSUB1500:POKE7680+C+B+TV,160:
NEXT
14110 IFHB=1THENHB=0:RETURN
14111 FORT=1TOWZ*2
14112 POKE36875,140+(T*2):FORS=1TO50
:NEXTS:POKE36875,0:FORS=1TO50
14113 NEXTS:NEXTT
14120 IFHB=1THENHB=0:RETURN
14121 GOTO10
```



Los chinos

C-64

Rafael Alberto García Medina
Carretera de Granada, nº 4, 3C
Guadix (Granada)

Este programa está basado en el tradicional juego de los chinos que tanto se juega en los bares con los amigotes.

Se juega contra el ordenador.

A este programa le tengo mucho, muchísimo cariño ya que con él he aprendido a dominar el juego perfectamente para más tarde ganarle a mis compañeros.

El juego es el siguiente:

El ordenador te desafía con la pregunta "¿a la de cuántas partidas jugamos?". Tú fijas libremente el número de partidas a jugar.

Seguidamente te pregunta "¿Cuántos chinos tienes supuestamente en tu mano?". Tú debes de contestarle con un número comprendido entre 0 y 3 inclusive.

Más tarde te dice "¿Cuántos pides?". Tú tienes que contestar con el número que crees que 'tiene él en su mano' sumado con el que tú tienes en la mano.

Es muy interesante porque una partida pides tú primero y en la siguiente te pide el ordenador.

Gana el que llegue antes a las partidas fijadas por el jugador al principio del juego.

PARECE COMPLICADO PERO EN LA PRACTICA ES MUY FACIL.

¡¡Ojo!! Cuidado con enfadarse con el ordenador o con picarse en el juego. Os digo esto porque lo sé por experiencia.

NO INTENTEIS ENGAÑARLO PORQUE OS LLEVAREIS UNA GRAN SORPRESA.

¡¡Animáos y desafiad a la máquina!!

```
10 REM ** LOS CHINOS **
20 POKE53280,0:POKE53281,13:Z=0:K=0
25 PRINT"[BLK][CLR]":INPUT"[4CRSRD]A
[SPC]LA[SPC]DE[SPC]CUANTAS[SPC]PARTI
DAS[SPC]JUGAMOS";W:FORJ=1TOW:PRINT"[BLK]
[CLR]"
26 PRINTTAB(15)"[CRSRD][RVSON]LOS[SPC]
CHINOS"
30 PRINT"[BLU]":PRINT"[RVSON]TU:"Z"P
UNTOS""[16CRSRR][RVSON]YO:"K"PUNTOS"
```

```
35 IFZ=WTHENPRINT"[RVSON][SPC]ME[SPC]
HAS[SPC]GANADO[SPC]POR";Z-K"PARTIDAS
[SPC]ENHORABUENA!!!":GOTO400
36 IFK=WTHENPRINT"[RVSON][SPC]JA,JA,
JA[SPC]TE[SPC]HE[SPC]GANADO[SPC]POR"
K-Z"PARTIDAS!!!":GOTO400
40 PRINT"[BLK][CRSRD]"
60 INPUT"CUANTOS[SPC]CHINOS[SPC]TIEN
ES[SPC]EN[SPC]LA[SPC]MANO";T
62 IFT>3ORT<0THENPRINT"[RVSON][SPC]T
RAMPOSO,MENTIROSILLO[SPC]!":GOTO60
63 INPUT"CUANTOS[SPC]PIDES[SPC]FORAS
TERO";P
70 IFP>6ORP<TTHENPRINT"[RVSON][SPC]A
SI[SPC]NUNCA[SPC]ACERTARAS[SPC]FORAS
TERO[SPC]":GOTO63
80 M=INT(RND(1)*4)
81 X=INT(RND(1)*4)
83 S=M+T
85 L=M+X
87 IFL=PORL<TTHEN81
88 IFL>T+3THEN81
90 PRINT"PUES[SPC]YO[SPC]PIDO";L
100 FORG=1TO1200:NEXTG
110 IFP=STHENPRINT"BIEN[SPC]COLEGA.T
U[SPC]GANAS.[SPC]TENGO:"M"CHINOS!!!"
:Z=Z+1
120 IFL=STHENPRINT"GANO[SPC]YO[SPC]C
OLEGA,PORQUE[SPC]TENGO:"M"CHINOS!!!"
:K=K+1
130 IFP<>SANDL<>STHENPRINT"NINGUNO[SPC]
HEMOS[SPC]ACERTADO!""YO[SPC]TENGO"M"
CHINOS"
140 PRINT"TOTAL[SPC]DE[SPC]CHINOS=";
S
145 IFZ=WORK=WTHEN400
150 PRINT"[2CRSRD]PULSA[SPC]UNA[SPC]
TECLA[SPC]PARA[SPC]CONTINUAR"
160 GETA$:IFA$=""THEN160
200 PRINT"[CLR]":PRINTTAB(15)"[CRSRD]
[RVSON]LOS[SPC]CHINOS"
210 PRINT"[BLU]":PRINT"[RVSON]TU:"Z"
PUNTOS""[16CRSRR][RVSON]YO:"K"PUNTOS
"
220 M=INT(RND(1)*4)
230 X=INT(RND(1)*4)
250 L=M+X
260 PRINT"[BLK]AHORA[SPC]YO[SPC]PIDO
...";L
270 FORG=1TO1200:NEXTG
280 INPUT"CUANTOS[SPC]CHINOS[SPC]TIE
NES[SPC]EN[SPC]LAS[SPC]MANOS";T
290 IFT>3ORT<0THENPRINT"[RVSON][SPC]
TRAMPOSETE,MENTIROSILLO[SPC]":GOTO28
0
295 S=M+T
300 INPUT"CUANTOS[SPC]PIDES[SPC]FORA
STERO";P
310 IFP>6ORP<0THENPRINT"[RVSON][SPC]
ASI[SPC]NUNCA[SPC]ACERTARAS[SPC]VILL
ANO[SPC]!":GOTO300
315 IFP=LTHENPRINT"[RVSON][SPC]YO[SPC]
SI[SPC]FUERA[SPC]TU[SPC]NO[SPC]DIRIA
[SPC]ESE[SPC]NUMERO[SPC]O.K?":GOTO30
0
320 IFP=STHENPRINT"BIEN[SPC]COLEGA.T
U[SPC]GANAS.[SPC]TENGO:"M"CHINOS!!!"
:Z=Z+1
330 IFL=STHENPRINT"GANO[SPC]YO[SPC]C
OLEGA,PORQUE[SPC]TENGO:"M"CHINOS!!!"
:K=K+1
340 IFP<>SANDL<>STHENPRINT"NINGUNO[SPC]
HEMOS[SPC]ACERTADO!""YO[SPC]TENGO"M"
CHINOS"
350 PRINT"TOTAL[SPC]DE[SPC]CHINOS=";
S
355 IFZ=WORK=WTHEN400
360 PRINT"[2CRSRD]PARA[SPC]CONTINUAR
```



```
[SPC]PULSA[SPC]UNA[SPC]TECLA"
370 GETA$:IFA$=""THEN370
390 NEXTJ
400 IFZ=WTHENPRINT"[RVSON][CRSRD]ENH
ORABUENA!!!TU[SPC]GANAS[SPC]POR";Z-K
"PARTIDAS!!!"
405 IFK=WTHENPRINT"[RVSON][CRSRD][SPC]
JA,JA,JA[SPC]TE[SPC]HE[SPC]GANADO[SPC]
POR"K-Z"PARTIDAS!!!"
408 PRINT"[3CRSRD]JUGAMOS[SPC]LA[SPC]
REBANCHA?[SPC](S/N)"
410 GETA$:IFA$="S"THEN10
420 IFA$<>"N"THEN410
```



¿Utiliza sólo un

Utilizar un COMMODORE 64 únicamente para jugar, es como pedirle a Albert Einstein que nos resuelva la raíz cuadrada de cuatro.

El cerebro del ordenador apenas funciona.

Para aprovecharlo al máximo, lo que Vd. necesita son programas interesantes (archivo de documentos, educación interactiva, estimulantes juegos o proceso de textos) y lógicamente los periféricos adecuados:

— La unidad de disco Commodore, un almacenaje de datos muy rápido, fiable y de gran capacidad.

— La unidad de cassette Commodore, una forma económica de introducir y almacenar programas.

— El monitor en color Commodore, para

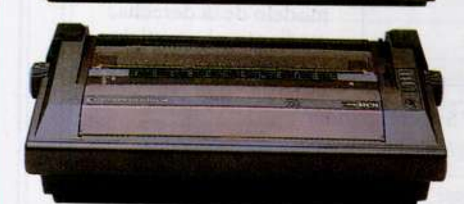
MICROELECTRONICA Y CONTROL c/ Valencia, 49-53. 08015 Barcelona. c/ Princesa, 47, 3.° G. 28008 Madrid.
UNICO REPRESENTANTE DE COMMODORE EN ESPAÑA



COMMODORE MPS 801
Impresora de matriz



COMMODORE MCS 801
Impresora cuatro colores



COMMODORE DPS 1101
Impresora de margarita



COMMODORE 1520
Printer plotter cuatro colores

COMMODORE 1541
Unidad de disco 170 K

COMMODORE 1530
Unidad de cassette





COMMODORE 1701
Monitor con color y sonido

JOYSTICKS
PADDLES
SOFTWARE Y LIBROS

COMMODORE COMPUTER Advierte que al comprar uno de sus productos exija la garantía de MICROELECTRONICA Y CONTROL, S.A., única válida para todo el territorio nacional.

10% de su cerebro?

lograr que los textos y gráficos sean más claros y vivos que en un televisor.

– 3 impresoras y un printer-plotter, para copias en papel, en blanco y negro o color.
– Los joystick y paddles, para conseguir que los juegos sean más excitantes.

Sea exigente. Si su cerebro, el de su COMMODORE 64, puede darle 100, no se conforme con 10.



la emoción del futuro



Puzzle

VIC-20
SUPEREXPANDER

Manuel Lucena López
Cuesta de Prieto, 4
Baeza (Jaén)

El puzzle consiste en ordenar según el modelo de la derecha, mediante el joystick, el dibujo de la izquierda que se desordena al azar. Se necesita Superexpander.

```
10 DATA81,40,85,113,73,41,87,30,105,
74,86,75,95,98,94,58,122
11 DATA32,76,59,31,100,78,79,88,80,7
7,100,109,32,76,46,122,32
12 DATA125,123,27,107,64,115,29,108,
102,104,92,116,117,97,160
20 FORI=0TO6:FORH=0TO6
30 READA(H,I):NEXT:NEXT
40 PO=7841
42 PRINT"[CLR]":FORI=38400TO38905:PO
KEI,6:NEXT
44 FORI=7680TO7688:POKEI,42:POKEI+10
,42:NEXT:FORI=7680TO7856STEP22:POKEI
,42:POKEI+10,42
45 NEXT
46 FORI=7856TO7864:POKEI,42:POKEI+10
,42:NEXT:FORI=7688TO7864STEP22:POKEI
,42:POKEI+10,42
48 NEXT:FORI=0TO6:FORH=0TO6:POKE7703
+H+I*22,A(H,I):POKE7713+H+I*22,A(H,I
):NEXT:NEXT
50 FORNN=1TO400
60 J=INT(RND(1)*4)+1
70 IFJ=2THENJ=-1
80 IFJ=3THENJ=22
90 IFJ=4THENJ=-22
100 IFPEEK(PO+J)=42THENGOTO60
110 POKEPO,PEEK(PO+J):POKEPO+J,160:P
O=PO+J
120 IFNN>400THEN140
130 NEXTNN
140 PRINT"[10CRSRD]CON[SPC]TU[SPC]JO
YSTICK[SPC]HAZ"
150 PRINT"EL[SPC]PUZZLE"
160 J=RJOY(0)
170 S=(((JAND4)=4)-((JAND8)=8))+(((J
AND1)=1)*22)-(((JAND2)=2)*22)
175 IFS=23ORS=21ORS=-23ORS=-21THEN16
0
180 IFPEEK(PO+S)=42THEN160
190 POKEPO,PEEK(PO+S):PO=PO+S:POKEPO
,160
200 FORI=0TO6:FORH=0TO6
210 IFPEEK(7703+H+I*22)<>A(H,I)THEN2
40
220 NEXT:NEXT
230 PRINT"FELICIDADES":END
240 FORI=1TO100:NEXT:GOTO160
```



El laberinto

C-64

Rafael Alberto García Medina
Carretera de Granada, nº 4, 3C
Guadix (Granada)

Es un juego de habilidad, rapidez y de mucha vista.

Se juega desde el teclado (mirar presentación del juego).

Te encuentras perdido en un laberinto en el que sólo sobreviven los más astutos. Tu misión es la de salir por las salidas que hay en la parte inferior de la pantalla cosa que no es nada fácil.

Además tienes que luchar con el tiempo que se visualiza en la pantalla ya que si sobrepasa los 200 segundos y aún no has encontrado la salida serás hombre muerto.

¡Ojo! MUCHO CUIDADO DE NO CHOCAR CONTRA LOS MUROS ya que se encuentran electrizados y si topas con ellos morirás electrocutado.

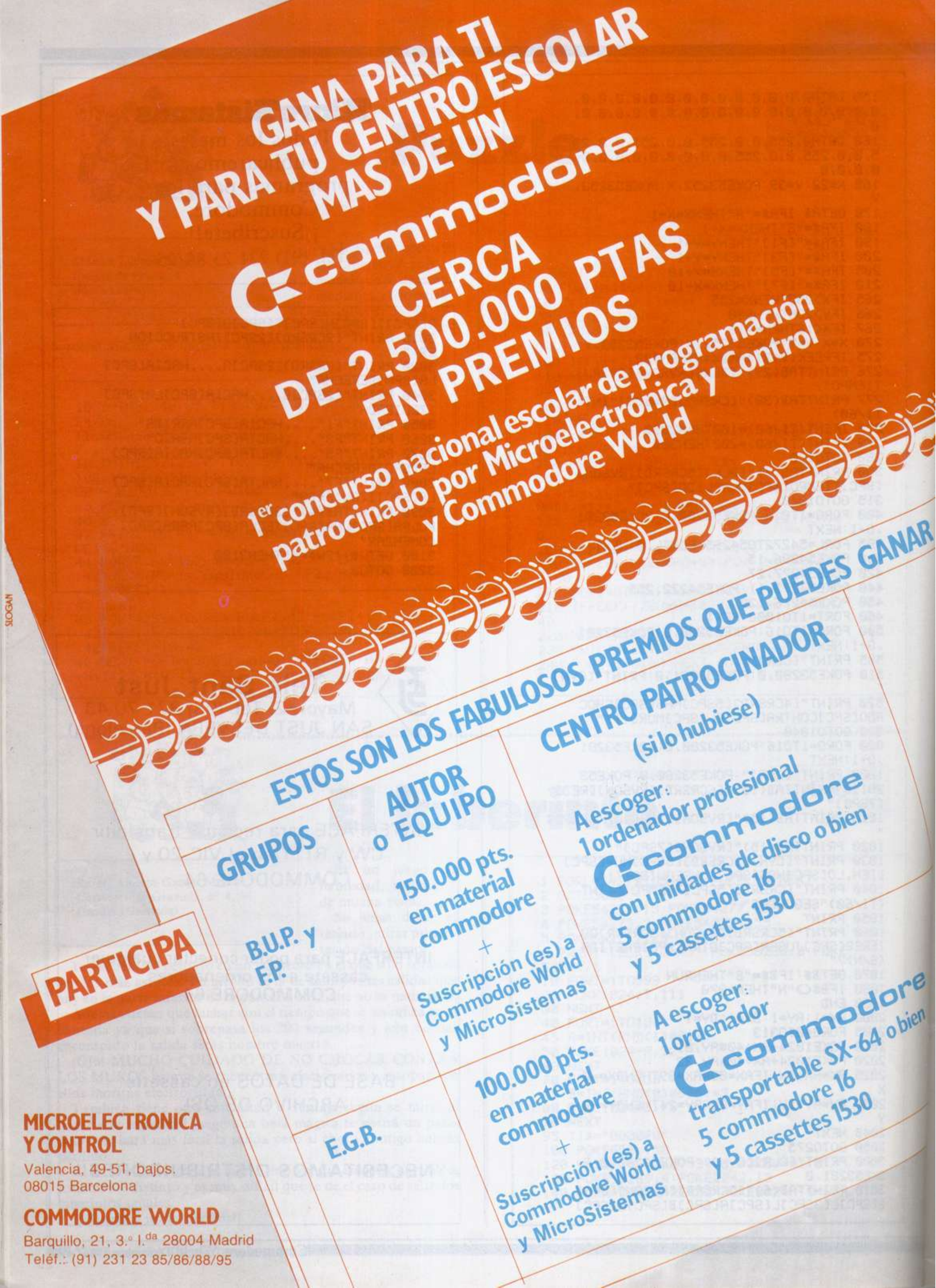
También tiene otra dificultad o ventaja según se mire: al llegar a la mitad del juego una bola mágica te abrirá un pasadizo y te hará más fácil la salida pero si choca contigo habrás perdido.

El laberinto no resulta monótono ya que en cada partida es totalmente distinto y es muy difícil que se dé el caso de salir dos laberintos iguales.

¡¡¡Suerte y no os perdáis!!

```
1 FORL=54272TO54295:POKEL,0:NEXT
2 GOTO3000
3 POKE54296,15:POKE54277,255
4 POKE54273,3:POKE54272,255
5 POKE54276,129
8 PRINT"[BLK][CLR]":POKE53280,0:POKE
53281,1
10 FORT=1TO999
20 POKE1024+T,111
30 NEXT
40 FORT=1TO100
45 A=INT(RND(1)*999)
50 POKE1024+A,32
60 NEXT
70 FORT=1TO130
75 B=INT(RND(0)*996)+3
80 POKE1024+B,76
90 NEXT
95 TI$="000000"
100 POKE650,150
120 V=53248:POKEV+34,3:POKEV+41,2
130 POKE53269,4:POKE2042,13
140 FORN=0TO62:READQ:POKE832+N,Q:NEX
T
```

```
150 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0
160 DATA0,255,0,0,255,0,0,255,0,0,25
5,0,0,255,0,0,255,0,0,0,0,0,0,0,
0,0,0,0
165 X=22:Y=39:POKE53252,X:POKE53253,
Y
170 GETA$:IFA$="A"THENX=X-1
180 IFA$="S"THENX=X+1
190 IFA$="[F1]"THENY=Y-1
200 IFA$="[F3]"THENY=Y+1
205 IFA$="[F5]"THENX=X+10
210 IFA$="[F7]"THENX=X-10
265 IFX>255THENX=255
266 IFY>236THEN800
267 IFX<16THENX=16
270 X=X:Y=Y:POKE53252,X:POKE53253,Y
275 IFPEEK(V+31)ANDX=XTHEN400
276 PRINTTAB(29)"[BLK][RVSON][CRSRU]
TIEMPO";
277 PRINTTAB(30)"[CRSRU][RVSON]"INT(
TI/60)
278 IFINT(TI/60)=100THEN2000
279 IFINT(TI/60)=200THEN300
280 GOTO170
300 PRINTTAB(7)"[BLK][5CRSRD][RVSON]
[SPC]TIEMPO[SPC]CUMPLIDO[SPC]"
315 GOTO1040
400 FORG=1TO16:POKE53280,G:POKE53281
,G+1:NEXT
405 FORL=54272TO54295:POKEL,0:NEXT
410 POKE54296,15
420 POKE54277,27
440 POKE54273,11:POKE54272,255
450 POKE54276,129
460 FORT=1TO1000:NEXT
500 FORG=1TO16:POKE53280,G:POKE53281
,G+1:NEXT
505 PRINT"[CLR]"
510 POKE53280,0:POKE53281,0:PRINT"[CYN]
"
520 PRINT"[4CRSRD][5SPC]HAS[SPC]CHOC
ADO[SPC]CONTRA[SPC]EL[SPC]MURO!!!"
530 GOTO1040
800 FORG=1TO16:POKE53280,G:POKE53281
,G+1:NEXT
1000 PRINT"[CLR]":POKE53280,0:POKE53
281,0:PRINTTAB(15)"[6CRSRD][RVSON][RED]
[7SPC]"
1010 PRINTTAB(15)"[RVSON][2SPC]O.K[2SPC]
"
1020 PRINTTAB(15)"[RVSON][7SPC]"
1030 PRINT"[CYN][2CRSRD][5SPC]MUY[SPC]
BIEN.LO[SPC]HAS[SPC]CONSEGUIDO!!!"
1040 PRINT"[CRSRD][5SPC]TIEMPO:";INT
(TI/60)"SEGUNDOS"
1050 PRINT
1060 PRINT"[5CRSRD][RVSON][5CRSRR]QU
IERES[SPC]JUGAR[SPC]OTRA[SPC]PARTIDA
(S/N)?"
1070 GETB$:IFB$="S"THENRUN
1080 IFB$<>"N"THEN1070
1090 END
2000 AX=1:AY=1:DX=1:DY=1
2005 FORG=0TO313
2010 POKE1024+AX+40*AY,81
2020 POKE1024+AX+40*AY,32
2025 AX=AX+DX:IFAX=0ORAX=39THENDX=-D
X
2030 AY=AY+DY:IFAY=0ORAY=24THENDY=-D
Y
2040 NEXT
2050 GOTO275
3000 PRINT"[CLR][CYN]":POKE53280,0:P
OKE53281,0
3010 PRINTTAB(6)"[5CRSRD][RVSON][SPC]
E[SPC]L[3SPC]L[SPC]A[SPC]B[SPC]E[SPC]
R[SPC]I[SPC]N[SPC]T[SPC]O[SPC]"
3015 PRINT"[2CRSRD][2SPC]INSTRUCCION
ES:"
3020 PRINT"[CRSRD][2SPC]A....HACIA[SPC]
LA[SPC]DERECHA"
3040 PRINT"[2SPC]S....HACIA[SPC]LA[SPC]
IZQUIERDA"
3050 PRINT"F1'....HACIA[SPC]ARRIBA"
3060 PRINT"F3'....HACIA[SPC]ABAJO"
3070 PRINT"F5'....SALTA[SPC]HACIA[SPC]
LA[SPC]DERECHA"
3080 PRINT"F7'....SALTA[SPC]HACIA[SPC]
LA[SPC]IZQUIERDA"
3090 PRINTTAB(4)"[3CRSRD][RVSON][SPC]
PULSA[SPC]UNA[SPC]TECLA[SPC]PARA[SPC]
COMENZAR"
3100 GETW$:IFW$=""THEN3100
3200 GOTO3
```



GANA PARA TI
Y PARA TU CENTRO ESCOLAR
MAS DE UN
commodore
CERCA
DE 2.500.000 PTAS
EN PREMIOS
1er concurso nacional escolar de programación
patrocinado por Microelectrónica y Control
y Commodore World
ESTOS SON LOS FABULOSOS PREMIOS QUE PUEDES GANAR
GRUPOS
AUTOR o EQUIPO
CENTRO PATROCINADOR (si lo hubiese)
B.U.P. y F.P.
150.000 pts. en material commodore + Suscripción (es) a Commodore World y MicroSistemas
A escoger: 1 ordenador profesional commodore con unidades de disco o bien 5 commodore 16 y 5 cassettes 1530
E.G.B.
100.000 pts. en material commodore + Suscripción (es) a Commodore World y MicroSistemas
A escoger: 1 ordenador commodore transportable SX-64 o bien 5 commodore 16 y 5 cassettes 1530
PARTICIPA
MICROELECTRONICA Y CONTROL
Valencia, 49-51, bajos.
08015 Barcelona
COMMODORE WORLD
Barquillo, 21, 3.º I.da 28004 Madrid
Teléf.: (91) 231 23 85/86/88/95



Pintor Loco



Gabriel Torres Rosselló (15 años)
Avda. Principes de España, 27 - 1º
(971) 54 63 95
ALCUDIA (Mallorca)

Un pintor, representado por un rodillo, debe pintar la fachada de una casa en un plazo determinado. Sin embargo, por la casa corre una araña muy peligrosa, que si pica al pintor lo hace polvo y tiene que volver a empezar los trabajos. Igualmente ocurre si no termina de pintar dentro del plazo.

El rodillo pinta con A, D, F1 y F7. La araña se mueve ella solita.

Si se quiere hacer el juego más difícil, se puede quitar la última línea de DATA, con lo que se reduce el tiempo. El recorrido de la araña se puede variar cambiando los DATA.

El programa ocupa toda la memoria del VIC, pero si se dispone de ampliación se pueden añadir sonidos y dejar el juego más atractivo.

Variables: R y R1 : Rodillo
A1 y A2: Araña
X e Y : Movimiento araña
Z : Movimiento rodillo
N : Puntuación

Descripción: Líneas

1-8	Presentación
10-64	Edición caracteres
200-229	Dibujo casa
300	Variables
400-430	Movimiento rodillo
520-610	Movimiento araña
1000	Subrutina pintura
1500-1550	Subrutina rodillo
2000	Subrutina restituir pintura paso araña
3000	Picadura araña
3100	Final juego cuando termina pintura
3300	Tiempo acabado
3500	Puntuación
5000-5150	DATAS movimiento araña

```
0 REM PINTOR LOCO
1 PRINT"[CLR]":PRINT"[RVSON]****[SPC]
PINTOR[2SPC]LOCO[SPC]****[RVSOFF]"
2 PRINT"[BLK][2CRSRD]1.[SPC]SE[SPC]T
RATA[SPC]DE[SPC]PINTAR[4SPC]LA[SPC]F
ACHADA[SPC]DE[2SPC]UNA[4SPC]CASA."
3 PRINT"[CRSRD]2.[SPC]A,D,F1[SPC]Y[SPC]
F7[SPC]MUEVEN[4SPC]EL[SPC]RODILLO[SPC]
DEL[SPC]PIN[4SPC]TOR.[13SPC][COMMT]"

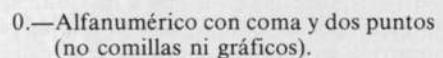
4 PRINT"[CRSRD]3.[SPC]CUIDADO[SPC]CO
N[SPC]EL[SPC]BI-[4SPC]CHO[SPC]QUE[SPC]
CORRE[2SPC]POR[4SPC]LA[SPC]CASA."
6 PRINT"[3CRSRD][RED][15SPC]G.T.R."
8 FORI=1TO8000:NEXT
10 POKE52,28:POKE56,28
20 FORT=7168TO7679:POKET,PEEK(T+2560
0):NEXT
30 POKE36869,255
40 READA:IFA=-1THEN200
50 FORT=0TO7:READB:POKE7168+A*8+T,B:
NEXT
60 GOTO40
61 DATA0,255,255,255,255,255,255,255
,255
62 DATA28,255,255,3,3,255,255,24,24
63 DATA30,36,24,255,60,255,60,255,24

64 DATA-1
200 PRINT"[CLR]":POKE36879,8
208 PRINT"[HOM][CRSRD][6CRSRR][WHT][RVSON]
[COMMD][COMMF][5CRSRR][COMMD][COMMF]
"
211 PRINT"[6CRSRR][RVSOFF][CYN]@@[5CRSRR]
@@"
212 PRINT"[6CRSRR]@@[5CRSRR]@@"
213 PRINT"[RVSON][RED][4CRSRR][13COMM+]
"
214 PRINT"[RVSON][RED][3CRSRR][15COMM+]
"
215 PRINT"[RVSON][RED][2CRSRR][COMM+]
[SHIFT£][COMM*][COMM+][SHIFT£][COMM*]
[5COMM+][SHIFT£][COMM*][COMM+][SHIFT£]
[COMM*][COMM+]"
216 PRINT"[RVSON][RED][CRSRR][2COMM+]
[YEL][COMMG][COMMM][RED][COMM+][YEL]
[COMMG][COMMM][RED][5COMM+][YEL][COMMG]
[COMMM][RED][COMM+][YEL][COMMG][COMMM]
[RED][2COMM+]"
217 PRINT"[RVSON][BLK][COMM+][YEL][COMMM]
[SPC][SHIFTL][SHIFT@][SPC][SHIFTL][SHIFT@]
[5SPC][SHIFTL][SHIFT@][SPC][SHIFTL][SHIFT@]
[SPC][COMMG][BLK][2COMM+]";
218 PRINT"[RVSON][COMM+][YEL][COMMM]
[7SPC][SHIFTO][COMMT][SHIFTP][7SPC][COMMG]
[BLK][2COMM+]";
219 PRINT"[RVSON][COMM+][YEL][COMMM]
[7SPC][COMMG][SPC][COMMM][7SPC][COMMG]
[BLK][2COMM+]";
220 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][SHIFTO][SHIFTP][SPC][SHIFTO][SHIFTP]
[SPC][COMMG][SPC][COMMM][SPC][SHIFTO]
[SHIFTP][SPC][SHIFTO][SHIFTP][SPC][COMMG]
[BLK][2COMM+]";
221 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][COMMG][COMMM][SPC][COMMG][COMMM]
[SPC][SHIFTO][COMMT][SHIFTP][SPC][COMMG]
[COMMM][SPC][COMMG][COMMM][SPC][COMMG]
[BLK][2COMM+]";
222 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][SHIFTL][SHIFT@][SPC][SHIFTL][SHIFT@]
[SPC][SHIFTL][COMM@][SHIFT@][SPC][SHIFTL]
[SHIFT@][SPC][SHIFTL][SHIFT@][SPC][COMMG]
[BLK][2COMM+]";
223 PRINT"[RVSON][COMM+][YEL][COMMM]
[17SPC][COMMG][BLK][2COMM+]";
224 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][SHIFTO][SHIFTP][SPC][SHIFTO][SHIFTP]
[SPC][SHIFTO][COMMT][SHIFTP][SPC][SHIFTO]
[SHIFTP][SPC][SHIFTO][SHIFTP][SPC][COMMG]
[BLK][2COMM+]";
225 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][COMMG][COMMM][SPC][COMMG][COMMM]
[SPC][COMMG][SPC][COMMM][SPC][COMMG]
[COMMM][SPC][COMMG][COMMM][SPC][COMMG]
[BLK][2COMM+]";
226 PRINT"[RVSON][COMM+][YEL][COMMM]
[SPC][SHIFTL][SHIFT@][SPC][SHIFTL][SHIFT@]
[SPC][COMMG][SPC][COMMM][SPC][SHIFTL]
```

(Pasa a pág. 65.)

Rutinas Basic (2)

Cursor



Jordi SASTRE

Bajo el nombre de CURSOR se esconde una potente rutina para las entradas de datos.

La instrucción INPUT del Basic tiene ciertos inconvenientes para cuando se desean programar entradas de datos muy controladas. Estos inconvenientes son principalmente dos: si se pulsa RETURN sin introducir ningún dato el programa queda cortado, y las teclas de control de cursor permanecen activas (cursor arriba, cursor abajo, borrado de pantalla, ...) con sus consiguientes peligros.

En una programación completa, tal como Contabilidad, Facturación o Control de Stocks, donde el ordenador es manejado generalmente por personas sin suficiente conocimiento informático, es necesario tomar precauciones para evitar que el operador no cometa fallos accidentales. Debe impedirse que pueda borrar la pantalla cuando el programa hace una pregunta, que el programa pueda interrumpirse accidentalmente, que escriba letras cuando lo que se pide son números, etc. Por otra parte debe dársele el mayor número de facilidades en las entradas de datos, como permitir el movimiento del cursor dentro de una zona destinada a respuesta, o efectuar el borrado e inserción de caracteres.

Una rutina que haga todo esto no es sencilla. Debe incorporar diversas funciones de edición, como el borrado de caracteres, el desplazamiento del cursor a derecha o izquierda sin alterar el texto, o la apertura del texto para la inserción de caracteres. Además es conveniente incluir un filtrado de teclas para que, por ejemplo, no se admitan caracteres alfabéticos cuando lo que se pide son números.

La rutina de cursor puede hacerse de muchas maneras, siendo todas ellas correctas. Aquí voy a describir una de ellas que cumple un mínimo de requisitos, y que posiblemente no es la mejor, pero al menos funciona. Toda entrada de datos en un ordenador debe tener un texto como pregunta a cada dato. Así pues esta rutina debe tener como variable de entrada una cadena con el texto de la pregunta (supongamos T$). La respuesta puede venir devuelta en otra variable de cadena, por ejemplo, B$, y en una numérica (B) para cuando el resultado sea numérico (basta con efectuar B=VAL(B$) antes de finalizar la rutina).

Utilizamos una variable de control, MX, en la cual se especifica la longitud máxima deseada en la respuesta y el tipo de filtrado. Ambas cosas se indican con la misma variable utilizando el punto decimal como separación. Por ejemplo MX= 15.2 indica que la entrada debe ser de un máximo de 15 caracteres con el filtro de tipo 2.

Los filtros que pueden efectuarse en una entrada de datos son muchos, pero quizá los más representativos o necesarios pueden definirse de la siguiente manera: un dato puede ser alfanumérico o numérico; el primero debe aceptar todo tipo de caracteres, mientras que el segundo únicamente digitos. Ampliando las posibilidades pueden haber dos tipos de filtro alfanumérico: uno que acepte la coma y los dos puntos y otro que no (las grabaciones en disco con PRINT# no permiten que las cadenas contengan comas o dos puntos). En ningún caso se aceptan las comillas, pues son conflictivas en el tratamiento de cadenas. Por su parte pueden facilitarse tres tipos de filtro numérico: uno que acepte, además de los digitos, el punto decimal y signo menos, un segundo que únicamente acepte el signo (para cantidades sin decimales) y un tercero que sólo acepte digitos (para códigos, por ejemplo). Con estos filtros cumplimos con la regla de oro enésima que dice: cuantas menos teclas pueda pulsar un operador, menos probabilidades hay de que se equivoque.

Por último puede indicarse también en la misma variable del filtro si se desea que la cadena de retorno venga complementada con espacios en blanco por la derecha hasta su longitud máxima (indicada en MX), o por el contrario que elimine los espacios de la derecha entregando la cadena a la longitud solo usada. Los espacios en blanco de la izquierda se eliminan siempre (son conflictivos en las operaciones de disco).

Resumiendo, los tipos de filtro utilizados por la rutina son los siguientes:

0.—Alfanumérico con coma y dos puntos (no comillas ni gráficos).
1.—Alfanumérico sin coma, dos puntos, comillas ni gráficos.
2.—Numérico. Sólo acepta: 0 1 2 3 4 5 6 7 8 9.
3.—Numérico. Sólo acepta: 0 1 2 3 4 5 6 7 8 9 - (cantidades enteras).
4.—Numérico. Sólo acepta: 0 1 2 3 4 5 6 7 8 9 - (para códigos).

En estos cinco tipos la cadena de retorno, B$, tiene la longitud ajustada al dato introducido. Los tipos del 5 al 9 son iguales a estos con la salvedad de que entregan B$ rellenado con espacios en blanco hasta su longitud máxima indicada en MX. El tipo 5 corresponde al 0, el 6 al 1, etc.

Explicación de la rutina

A primera vista la rutina parece larga y compleja. Realmente no es corta ni sencilla, pero merece un poco de explicación:

30002: La variable MX se desglosa en tres: X5 es el tipo de filtro (del 0 al 9). X0 también es el tipo de filtro, pero ignorando la longitud de salida (de 0 a 4). MY es la longitud máxima de la respuesta (parte entera de MX).

30009: Imprime la pregunta, un interrogante y una flecha marcando el final de la zona permitida para respuesta. Calcula en X1 la posición (byte de pantalla) actual del cursor.

30012: Dos variables de trabajo: X3 apunta al inicio de la zona de respuesta y X4 al final.

30015 y 30018: Bucle de espera de pulsación de tecla. Provoca el parpadeo del cursor hasta que se pulsa una tecla.

30021: Recoge el valor ASCII de la tecla pulsada, apaga el cursor y bifurca a 30063 si se ha pulsado RETURN (tecla de validación).

30024: Ejecuta el CURSOR-DERECHA.

30027: Ejecuta el CURSOR-IZQUIERDA.

30030 a 30036: Ejecuta el BORRADO-CARACTER (tecla INST/DEL).

30039 y 30042: Ejecuta la INSECCION-CARACTER (tecla INST/DEL con SHIFT).

30045: Ejecuta HOME (posicionarse al inicio de la zona).



30048: Ejecuta tecla CLR/HOME con SHIFT para posicionarse al final de la zona.

30051: Filtro principal: Comprueba que no se sobrepase el tamaño de la zona de respuesta y que el carácter pulsado esté comprendido entre los valores ASCII 32 y 96. No acepta las comillas (valor ASCII 34) ni los gráficos (valores mayores que 96).

30052: La coma y los dos puntos sólo se aceptan en tipo 0.

30054: El tipo 4 no acepta el signo menos.

30057: Los tipos 3 y 4 no aceptan punto decimal. Filtro numérico.

30060: Una vez aceptado el carácter se imprime y se incrementa el contador para después volver al bucle principal.

30063 a 30065: Aquí viene cuando se ha pulsado RETURN. Comprueba que como mínimo se haya escrito un carácter.

30066 y 30069: Recoge el contenido de la pantalla y lo asigna a B$.

30072: Convierte el interrogante en dos puntos y borra la flecha del final.

30075: Elimina los espacios en blanco iniciales.

30077: Si la longitud es menor que 10 obtiene el valor de la cadena en B.

30078: Si el tipo de filtro es de 5 a 9 complementa con espacios en blanco por la derecha.

30081: Retorno de la rutina.

Observaciones:

Esta rutina está escrita para una CPU de la serie 4000 o 8000. En la línea 30009 pueden apreciarse dos funciones PEEK. Lógicamente los valores indicados no sirven para otras CPUs, como el 700, el C-64 o el VIC-20. Para adaptar la rutina a estos ordenadores deben cambiarse los valores de los PEEKs: en lugar de 197 y 196 debe ponerse 201 y 200 en la serie 700, ó 210 y 209 en el C-64 y VIC-20.

Esta rutina utiliza algunas variables de trabajo, que deben conocerse para entrar en conflicto con ellas por ser utilizadas en el resto del programa. Al llamar a la rutina debe estar ya asignada la variable Z2$ con espacios en blanco; por ejemplo al principio del programa puede hacerse:

Z2$="": FOR X=1 TO 80 : Z2$=Z2$+" " : NEXT

Además de las variables de entrada/salida T$, MX, B$ y B, la rutina de cursor utiliza las siguientes:

- X5 — Tipo de filtro (de 0 a 9).
- X0 — Tipo de filtro (de 0 a 4).
- MY — Longitud máxima para la respuesta.
- X — Contador general, utilizado en varios puntos.
- X1 — Puntero al byte de pantalla en curso.
- X2 — Variable temporal que guarda el carácter durante el parpadeo del cursor.
- X3 — Puntero al inicio de la zona de respuesta (límite izquierdo de X1).
- X4 — Puntero al final de la zona de respuesta (límite derecho de X1).
- X9 — Variable auxiliar utilizada en 30066 y 30069.
- R$ — Carácter correspondiente a la tecla pulsada.
- XR — Valor ASCII de R$.

```
10 REM          *************
20 REM          *           *
30 REM          *  CURSOR   *
40 REM          *           *
50 REM          *************
60 :
100 REM DATOS DE ENTRADA
110 REM ----------------
120 REM  T$=MENSAJE PREGUNTA
130 REM  MX=LONGITUD MAXIMA DE LA RE
SPUESTA (PARTE ENTERA)
140 REM     TIPO DE ENTRADA (PARTE D
ECIMAL)
150 REM
160 REM DATOS DE SALIDA
170 REM ---------------
180 REM  B$=RESPUESTA INTRODUCIDA AN
TES DE PULSAR RETURN
190 REM B=RESPUESTA (VALOR NUMERICO
DE B$)
200 REM
210 REM OBSERVACIONES
220 REM -------------
230 REM  Z2$ DEBE CONTENER ESPACIOS
EN BLANCO.POR EJEMPLO, PUDEMOS EJECU
TAR
240 REM         AL PRINCIPIO:
250 REM         10 Z2$="":FORI=1TO80:Z2
$=Z2$+"[SPC]":NEXT
260 REM
270 REM
999 :
30000 REM*** CURSOR 4.1 *************
***
30001 REM*** I: T$ MX INT(MX)=LEN DE
C(MX)=TIPO
30002 X5=INT((MX-INT(MX))*10+.5):X0=
(X5+5*(X5>4)):MY=INT(MX)
30009 PRINTT$"?[SPC]"SPC(MY)"←[CRSRU]
":PRINTT$"?[SPC]";:X1=256*PEEK(197)+
PEEK(196)+LEN(T$)+2
30012 X3=X1:X4=X1+MY-1
30015 X2=PEEK(X1):POKEX1,X2+128:FORX
=1TO50:GETR$:IFR$<>""THEN30021
30018 NEXT:POKEX1,X2:FORX=1TO50:GETR
$:IFR$=""THENNEXT:GOTO30015
30021 XR=ASC(R$):POKEX1,X2:IFXR=13TH
EN30063
30024 IFXR=29ANDX1<X4THEN30060
30027 IFXR=157ANDX1>X3THENX1=X1-2:GO
TO30060
30030 IFXR<>20ORX1=X3THEN30039
30033 FORX=X1-1TOX4-1:POKEX,PEEK(X+1
):NEXT:POKEX4,32:X1=X1-1:PRINT"[CRSRL]
";
30036 GOTO30015
30039 IFXR<>148ORPEEK(X4)<>32ORX1>X4
THEN30045
30042 FORX=X4TOX1+1STEP-1:POKEX,PEEK
(X-1):NEXT:POKEX1,32:GOTO30015
30045 IFXR=19THENX1=X3:PRINT:PRINT"[CRSRU]
"T$"[2CRSRR]";:GOTO30015
30048 IFXR=147THENX1=X4:PRINT:PRINT"
[CRSRU]"T$SPC(MY+1);:GOTO30015
30051 IF(X1>X4)OR(XR>96ORXR<32ORXR=3
4)THEN30015
30052 IFX0>0AND(XR=58ORXR=44)THEN300
15
30054 IFX0=4ANDXR=45THEN30015
30057 IF(X0>2ANDXR=46)OR(X0>1AND(XR<
45ORXR>57ORXR=47)ANDXR<>32)THEN30015

30060 PRINTR$;:X1=X1+1:GOTO30015
30063 B$="":X=X4+1
30064 X=X-1:IFX<X3THEN30015
30065 IFPEEK(X)=32THEN30064
30066 FORX=XTOX3STEP-1:X9=PEEK(X):IF
X9<32THENB$=CHR$(X9+64)+B$:NEXT:GOTO
30072
30069 B$=CHR$(X9)+B$:NEXT
30072 PRINT:PRINT"[CRSRU]"T$":[SPC]"
SPC(MY)"[SPC]"
30075 IFLEFT$(B$,1)="[SPC]"THENB$=RI
GHT$(B$,LEN(B$)-1):GOTO30075
30077 B=0:IF MY<10THENB=VAL(B$)
30078 IFX5>4THENB$=LEFT$(B$+Z2$,MY)
30081 RETURN
```

Magia



La MAGIA son trucos, la MAGIA es divertida.

La MAGIA es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La MAGIA es una sección mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

MAGIA habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enviárnoslos comprobados, pues hay varios incorrectos. ¡Ah! y no nos mandéis trucos repetidos... ¡Listos!. Premios: Ver pág. 5.

NO AL REDO FROM START

Cada día son más los usuarios de CBM-64 y VIC-20, que realizan programas de aplicaciones técnicas, de gestión, educativas, etc... Por esta razón, creo que esta sección de MAGIA con sus pequeños trucos, tiene una gran importancia en el acabado de estos programas.

Os adjunto el listado de una subrutina que puede ser muy útil para evitar que el inoportuno "REDO FROM START", desfigure una pantalla, laboriosa de realizar, en la que el operador ha de introducir varios datos.

Esta subrutina utiliza la instrucción GET, en lugar del usual INPUT, pudiendo así analizar uno a uno los caracteres introducidos por teclado y colocarlos en el lugar de la pantalla que más nos interesa.

```
READY.
100 PRINT"[CLR]":X=12:Y=:GOSUB500:PRINTX1$;Y1$;"INTRO-
    DUCE NUMERO = "
110 GETA$:IFA$=""THEN110
120 IFASC(A$)=45ANDA1$=""THENA1$=A$:GOSUB610:GOTO110
130 IFASC(A$)=13THENA=VAL(A1$):AI$="":GOTO SALIR A
    CONTINUAR PROGRAMA
140 IFASC(A$)=20ANDLEN(A1$)<2THENA1$="":GOSUB600:GO-
    TO110
150 IFASC(A$)=20THENA1$=LEFT$(A1$,LEN(A1$)—1):GOSUB600
    :GOTO110
160 IFASC(A$)<48ORASC(A$)>57THENGOTO110
170 A1$=A1$+A$:GOSUB610:GOTO110
500 X$="[HOME][23 CRSR D]":Y$="[40 CRSR R]"
510 X1$=LEFT$(X$,X+1):Y1$=LEFT$(Y$,Y):RETURN
600 X=12:Y=22:GOSUB500:PRINTX1$;Y1$"[17 SPC]"
610 X=12:Y=22:GOSUB500:PRINTX1$;Y1$;A1$:RETURN
```

Alex Parellada Frouchtman
Angel Guimerá, 11-2º-2.ª
S. Cugat del Vallés
BARCELONA

REPETICION DE LINEAS

Algunas veces nos encontramos en un programa con líneas que son iguales y os mando un truco (muchos de vosotros ya lo conoceréis) que sirve para no repetir la línea. Ahí va:

Sólo hay que cambiar el número de línea y pulsar RETURN. Pero cuidando siempre de que si el número es por ejemplo 155 y queremos hacer la 156 no sólo debemos de cambiar el 5 por el 6, debemos cambiar todo el número. (Esto también sirve para líneas parecidas.)

José Manuel Mera Alonso
Buenos Aires, 2-2º C
VIGO (Pontevedra)

PROTECCION DE LISTADOS

1) Para proteger los listados del VIC.

```
1 REM (SHIFT L)
```

Cualquier listado encabezado con esta línea de programa será imposible de listar. Naturalmente, eliminando esta línea, el programa podrá ser listado, pero hay métodos para hacer invisible esta línea (que creo han salido en esta misma sección y por tanto no los repito) con lo cual toda persona extraña al programa no sabrá qué línea tiene que eliminar (aunque si sabe cómo trabaja el VIC-20 no le costará encontrarla).

2) POKE 808, 129 inutiliza las teclas RUN/STOP y RESTORE.

3) Para borrar una línea (o fila) de la pantalla:

```
POKE 781,X:SYS 60045
```

siendo X un nº entre 0 y 22 correspondiente a la fila que se quiere borrar.

Pere Busquets Congost
Rambla Llibertat, 16
17004 GIRONA

ERRORES EN DATAS

Os mando un pequeño programa que nos ayuda a encontrar los errores en las sentencias DATAS. Ahí va:

```
nnn10 A= 9999999999
nnn15 PRINT "(CLR)VD. Está en la línea data número (R)";
      PEEK (63) + 256 × PEEK(64)
nnn20 PRINT "La data de esta línea es (R)"; A$
nnn21 PRINT "(10 cursores abajo"
nnn30 IF VAL (A$) () INT (VAL(A$)) THEN PRINT
      "(R) usted ha confundido un punto con una coma"
nnn40 IF RIGHT$(STR$(VAL(A$)), LEN (A$)) () A$ THEN PRINT
      "(R) usted ha digitado una letra"
nnn50 INPUT A
nnn60 if a= 9999999999 THEN RETURN
nnn70 IF A () VAL (A$) THEN PRINT "(R) Data equivocada"
nnn80 PRINT "Pulse espacio para continuar"
nnn90 GET RE$: IF RE$ () " " TEEN nnn90
nn100 RETURN
```

Como habréis visto, este pequeño programa está preparado para ser una subrutina. Antes que nada, si el programa principal tiene la instrucción READ A, deberá ser cambiada por READ A$.

La línea nnn15 nos dice la línea DATA que está leyendo.

La línea nnn20 nos dice la data que está leyendo en ese momento.

La línea nnn30 nos dice, si, al escribir los datos, se nos ha colado alguna coma. OJO. Esta línea no servirá si trabajamos con DATAS fraccionarias o no enteras. Esta línea es muy fácil. Si un número no es igual a su parte entera, es porque tiene decimales.

La línea nnn40 nos avisa si hemos digitado una letra, como, por ejemplo, una "l" por un uno, o una "O" un Ø.

En la línea nnn50 nosotros le damos al ordenador el valor que creemos que tiene esa DATA, y si no coincide, nos contestará lo que aparece en la línea nnn70. Esto es muy útil cuando son dos personas. Cuando una dicta, la otra digita, hasta encontrar la DATA mal.

Pero, también hay otra forma de hacerlo. Como en la línea nnn20, nos va saliendo la DATA, cuando lleguemos al INPUT, si se quiere, basta con pulsar RETURN, para que salga la siguiente data.

Creo que este programa nos ahorra bastantes quebraderos de cabeza con las DATAs, sobre todo, a la hora de encontrar la que está mal.

Juan Martín
Las Palmas

EFECTOS ESPECIALES

Hola, gente, estoy aquí porque he descubierto una nueva MAGIA. Esta les va a encantar a los que hagan jueguitos con explosiones y esas cosas.

Bueno, esta es:

```
00001 PRINT "(CLR) DEMOSTRACION DEL MOVIMIENTO"
00002 P= 53265
00003 INPUT A
00004 IF A= -1 THEN END
00010 FOR V= Ø TO 99
00020 POKE P, 27
00030 GOSUB 00100
00040 POKE P, A
00050 GOSUB 00100
00060 NEXT V
00070 GOTO 1
00080 :
00100 FOR T= Ø TO 250: NEXT T
00110 RETURN
00120 :
```

Con esto se consigue que la pantalla haga efectos raros, según el valor que le demos a A (línea 00003), por ejemplo, podéis probar con éstos:

Cuando A=87, la pantalla se encoge.
Cuando A=79, la pantalla aparece y desaparece.
Cuando A=60, la pantalla parpadea con "basura".
Cuando A=88, la pantalla se mueve de arriba a abajo. Esto queda interesante con un ruido de explosión. Parece un terremoto.
Cuando A=255, lo mismo que cuando A=79, pero en negro.

"Na", que soy un genio.

La verdad que es muy interesante eso de hacer:

```
10 FOR X=OR TO 65000
20 PRINT X
30 POKE X, 34
40 NEXT
```

Lo malo es que cada dos por tres se queda "clavado", ¡¡hay un truco!!, pero luego hay que recoger el programa de nuevo. ¡¡¡Una lata!!!

Juan Martín
Las Palmas

ANTI-LIST

Os mando mi primera colaboración, que espero sea de utilidad para los demás compañeros. Lo que escribo es lo siguiente. Respecto a la función SQR(X), que se puede simplificar por esta pequeña instrucción:↑.5

También los siguientes POKEs:

Al listar un programa si escribimos POKE7774,34:POKE775,253, y escribimos LIST, el programa se autodestruye, desviando la dirección de memoria.

C. Luis Cerón
Plaza de España, 5-2º
Almendralejo (Badajoz)
Tengo 13 años.

FACTORIALES

Soy lector habitual de la revista Commodore World, poseo un CMB-64 desde hace algunos meses y es una pena que cuando John G. Kemeny y Thomas E. Kurtz idearon por primera vez el lenguaje BASIC pensarán que con una potencia de cálculo de 10 elevado a la 38 hubiera más que suficiente para cualquier aplicación.

Lo cual es falso en más de una ocasión, para solventar este inconveniente, he ideado un sistema el cual no ocupa más que una línea de programa, esto es:

```
8Ø IF a>1Ø THEN A=A/1Ø : H=H+1 : GOTO 8Ø
```

Para ilustrar su aplicación envío un pequeño programa que realiza el factorial de un número.

```
1Ø rem PROGRAMA DE FACTORIALES ENESIMOS
2Ø PRINT"[CLR] [14CRSRR] FACTORIALES [2CRSRD]"
3Ø A= : H=Ø: POKE5328Ø,6
4Ø INPUT" ? SU NUMERO ES"; N: PRINT" [CLR]"
5Ø IF N <Ø THEN PRINT" NO EXISTE FACTORIAL PARA N
     NEGATIVOS" :GOTO 2ØØ
6Ø IF N=Ø THEN PRINT" N!= 1" : GOTO 2ØØ
7Ø FOR X=1 TO N: A=A*X
8Ø IF A>1Ø THEN A=A/1Ø : H=H+1 : GOTO 8Ø
9Ø NEXT: IF H<9 THEN PRINT N;"!=";A*1Ø↑(H): GOTO 2ØØ
1ØØ PRINTN;"!=";A;"E";H
2ØØ PRINT TAB(5) "[5 CRSRD] PULSAR PARA..."
21Ø PRINTTAB(1Ø) " OTRO NUMERO...........1"
22Ø PRINTTAB(1Ø) " FIN DE TAREAS ..........2"
23Ø INPUT N : IF N=1 THEN 2Ø
24Ø END
```

Esto es todo por ahora. Espero haberos sido útil.

NOTA: Si se quita la línea 8Ø, se observará que el mayor factorial que se puede hacer es de 33.

José Rullo Batalla
Vilafranca, 2, Aptos. SOL IXENT 2
Comarruga (Tarragona)

TRUCOS EN IMPRESORA

Estos trucos son para el C-64 y son trucos de una sóla línea.

1º **TRUCO PARA RESOLVER UNA ECUACION DE SEGUNDO GRADO DE LA FORMA: $ax^2 + bx + c = 0$:**

Este truco es muy eficaz para los medianos estudiantes como yo y además ahorra muchísimo tiempo en la resolución.

Aparecerá en la pantalla:

?
??
??

En la primera interrogación tenemos que introducir el valor de **a**, en la segunda interrogación introduciremos el valor de **b** y en la tercera el valor de **c**. Mucho cuidado con intentar resolver ecuaciones sin solución dentro de los reales ya que entonces se generaría el mensaje de error:

? ILLEGAL QUANTITY ERROR IN 1.

El truco es el siguiente:

```
1 INPUTA,B,C:X=—B+SQR(B↑2—A*C*4):X=X/2*A:?X:Y=—B—SQR
(B↑2—A*C*4):Y=Y/2*A:?Y
```

2º **EFECTOS ESPECIALES:**

Este truco es digno de ver porque genera cosas muy bonitas en la pantalla que no explico porque... "Más vale una imagen que mil palabras".

Este es el truco:

```
1 N=N+1:?[CLR]":FORT=1Ø24TO2Ø25STEPN:POKET,81:NEXT:
GOTO1
```

Rafael Alberto García Medina, 17 años
Carretera de Granada, 4-3C. Teléf. 66 20 94
Bloque Medina Azahara
Guadix (GRANADA)

PRINTS MODIFICADOS

Queridos amigos de Commodore World, soy un Vic-adicto a vuestra revista. Os felicito por la revista, es estupenda.

Os escribo para mandaros dos programillas cortos para presentación de cadenas de caracteres. Si os gustan podéis publicarlos en la sección de Magia, si no en el cesto de los papeles estarán bien, no me enfado.

El primer programa coge la cadena y la presenta en pantalla, carácter a carácter, de derecha a izquierda y la centra. Como es para el Vic-20 la cadena tiene que ser de 1 a 22 caracteres para que funcione bien.

```
10C$ = "cadena":print" [2CRSRD] ":LC=Len(C$)
20 For X = 21 to (22-LC)/2 step-1:M=(22-X):IFM>LC Them M = LC
30 print tab (X) "[3CRSRU]" left $(C$,M)" [22spc]"
40 For L = 1 to 150:Mext L, X
```

Variando el valor 150 de la línea 40 las cadenas salen más lentamente o más rápidamente.

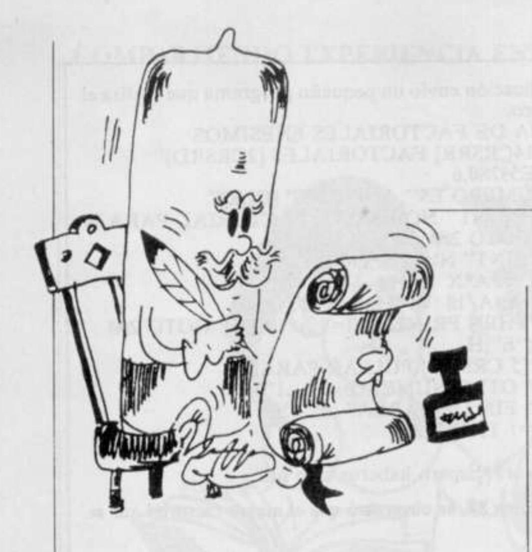
El segundo programa hace lo mismo que el primero pero entra la cadena de izquierda a derecha de la pantalla.

```
10C$="cadena":LC=Len(C$)
20ForX= to (22-LC)/2+LC:M=X:IFM>LCTHEN M=LC
30 print tab (X-M) "[2CRSRU] [22spc]" Right$(C$,M)
40 For L=1 to 150:Next L,X
```

Sin más me despido de vosotros hasta otra.

Santiago Luis Valdrés Allueva
Buenos Aires, 8
Zaragoza-15

CARTA BLANCA



DISCRIMINACION

Primero de todo os quiero echar una pequeña "bronca".

Soy asíduo lector de vuestra revista y la compro puntualmente todos los meses en el quiosco de mi barrio. *No soy suscriptor* y no es porque os tenga manía ni nada por el estilo sino porque ya me he cansado de que el cartero me doble la revista y la rasque con el borde del cajetín o de que la revista "vuele" del buzón gracias a algún vecino amigo de lo ajeno. (Esto son experiencias de antiguas suscripciones, no de vuestra revista). Por eso *si pago más cara* la revista que un suscriptor será por algún motivo y me creo con el mismo derecho que él a haceros consultas (¿no es así?). Por eso no me hace ninguna gracia que cuando os llamo por teléfono (sólo lo he hecho dos veces) lo primero que se me pregunta es: ¿eres suscriptor?, poniéndome "malas caras" (es un decir) en cuanto digo que no.

Este es un problema que ocurre, no sólo con vuestra revista sino con la mayoría (de cualquier tipo) de las que hay en el mercado. ¿Por qué esta discriminación con el lector no-suscriptor? Pensar en ello.

El otro motivo de esta carta es el de pediros un favor (qué paradoja ¿no?).

Tengo una impresora Admate DP-100 y me gustaría que publicáseis el programa que cambia los caracteres de control del Commodore (tengo un 64) a algo más legible ([3SPC], [5CRSRD]) ya sé que este programa lo usáis en una impresora Commodore para los listados de la revista pero creo que serviría con cualquiera o por lo menos intentaría adaptarlo a mis necesidades. Mi opinión general sobre la revista es bastante buena y alguno de los artículos son francamente interesantes ya que me están ayudando a conocer cada día un poco más las posibilidades de mi "64".

Al ordenador intento darle un uso "serio" (no lo compré para "matar marcianitos") y tengo prácticamente terminada una "base de datos" para el envío de "mailing" y en la que he estado trabajando algunas semanas.

No os doy más la lata.

Gracias anticipadas.

LUIS MARTIN MORIS
MATILDE DIEZ, 10
28002 MADRID

En primer lugar debemos aclarar que la pregunta "¿eres suscriptor?" se hace normalmente por costumbre, ya que las llamadas que recibimos suelen ser consultas sobre la suscripción, fecha de renovación, un número que se ha extraviado o cosas similares, y se hace para ir ganando tiempo al buscar en el ordenador la ficha para comprobar si los datos están todos correctos o existe algún problema. También estamos añadiendo a las fichas de los suscriptores sus números de teléfono cuando nos llaman, para que en el caso de que nos devuelva un número de la revista el cartero podamos comprobar si fue por error o cuál fue la causa.

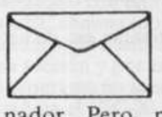
Nadie pone malas caras cuando se dice que el que llama no es suscriptor, no hacemos por tanto ninguna discriminación, y de hecho nos han llamado algunas veces lectores nuestros para que resolviésemos pegas con algunos programas de otras revistas que no les funcionaban y no se lo sabían resolver los otros. Nosotros intentábamos contestarles, pero algunas veces es muy difícil, ya que no conocemos los programas.

Respecto a la impresora, imaginamos que ya tienes el programa que hace que los open 4,4, los print #4, etc., funcionen con ella, y en caso contrario, tenemos preparado para enero o febrero un artículo sobre esto. Si lo que necesitas es el programa que cambia los caracteres gráficos de las impresoras Commodore por las siglas que utilizamos en la "clave para interpretar los listados", debo decirte que fue publicado en el número 6 de Club Commodore bajo el título de "los nuevos listados de club commodore", y nosotros sólo le hemos hecho unas pequeñas modificaciones y lo hemos compilado con ayuda del compilador Blitz para que su velocidad al "traducir" los listados sea mayor, ya que el tiempo es demasiado importante cuando se tienen que hacer cientos de listados.

En particular yo mismo tengo una Admate DP-80, y tengo que modificar este programa para obtener los símbolos de flechas arriba y a la izquierda del Commodore, ya que esta impresora no dispone de estos caracteres. También puedo comentarte que va fabulosamente bien con el procesador de texto "Easy Script" del Commodore 64 y que gracias a los cambios de idiomas de que dispone (8 idiomas en total), puedo utilizar aperturas de admiración, interrogación, acentos, etc., con ella.

Esperamos que si tienes cualquier pregunta nos escribas o llames por teléfono, y... no te molestes por la típica pregunta... es la costumbre, no por discriminar.

¿RESET O BORRADO?



—En la revista nº 3 (abril 84), pág. 38, se explica cómo añadir la tecla RESET a un ordenador. Pero, pregunto: Al añadir esta tecla y pulsarla ¿qué sucede con lo que haya en la memoria? ¿Se guarda o desaparece? Porque en el caso de que todo vuelva a estar como al principio, no veo qué ventaja tiene la tecla respecto a desconectar el ordenador.

—¿Cuáles son los límites de temperatura que tolera el C-64? (Funcionando normalmente, claro).

—¿Cuántos ordenadores existen en España? Y de ellos, ¿cuántos Commodore?

—En la feria INFORMAT-84, que formaba parte de EXPOTRONICA, el pasado mes de mayo, en Barcelona, pudimos ver un C-64 que hablaba. Quisiera saber cómo. Es decir, ¿se trata de un programa que aprovecha el generador de sonido del C-64, o había algo más?

Bien, por ahora nada más. Saludos.

JUAN MAURI I CANET
APTDO. DE CORREOS 4
LA ROCA (BARCELONA)

1. Los programas que hay en la memoria desaparecen. En realidad no desaparecen, pero el ordenador así lo cree. Los gráficos, por ejemplo, siguen tal y como estaban, al igual que los programas en lenguaje máquina.

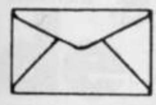
2. Los chips suelen funcionar bien hasta 50-60 grados. Hay que tener en cuenta que la temperatura externa del 64 no es la misma que la temperatura interna.

... Y SEAMOS

3. Desconocemos los datos exactos, porque el número aumenta día a día, así que no podemos darte una respuesta exacta.

4. Los hay de dos tipos, los que funcionan a base de programa y los que utilizan además un sintetizador de voz. Desconocemos cuál es el que viste.

ACLARACIONES



Les tengo que decir que al recibir el nº 9 de su revista me he llevado una sorpresa al no ver la "abeja" de metedura de pata del nº 8, puesto que hay un par de listados que no me dan resultado, por lo que les escribo a vuelta de correo para que en el próximo número de su revista me puedan aclarar el error si puede ser.

En el programa "Asterisco" de la pág. 38, al escribir la línea 2110 y al apretar la tecla Return me sale un mensaje de error todas las veces que lo he intentado, me parece que es debido a que es demasiado larga la línea, ya que si la escribo con 4 dígitos menos sí me entra, pero claro, supongo que luego ya no iría el programa.

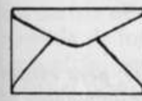
El segundo programa es el que hay en la pág. 34 "Laberinto" pues de la línea 230 a la 260 hay las instrucciones para mover el pac-man y sorpresa se me presentó al comprobar al acabar el programa que por mucho que aprietes esas teclas el pac-man no se mueve de sitio, pero en cambio sí que se mueve con el joystick.

Bien, esto es todo, espero que me lo aclaren en el próximo número, si puede ser, y les doy las gracias por adelantado.

CARLES BALAÑA MAGRANE
TIVOLI, 11-5 A
REUS (TARRAGONA)

Las preguntas que tengáis que hacer sobre programas de COLABORACIONES, es mejor que las hagáis a los propios autores, ya que para eso publicamos su dirección.

MAS MEMORIA



1.—Para pasar los 8 K del cartucho VIC-1110 a ROM, en lugar de a RAM, basta con desconectar el interruptor 1 y conectar los otros tres.

(Así: ↑↓↓↓)
1234

Para sumar 8 K RAM + 16 K RAM, insertando los dos cartuchos de ampliación, la posición es: desconectar el número 2 y conectar los otros tres. (↓↑↓↓).

Para sumar 8 K RAM a los 3 K de la versión base, se desconecta únicamente el 4. (↓↓↓↑).

Pero ¿qué ha de hacerse para obtener los 32 K que pueden lograrse con los tres cartuchos de ampliación, de modo que resulte efectiva la suma?

2.—No encuentro en ninguna parte el significado **exacto** de este mensaje de error:

```
? FILE DATA
ERROR IN....
```

¿Podéis decírmelo?

3.—Ejecutad este programa con una sola instrucción:

```
10 FOR I = 7 TO 28 : PRINT I * I, ↑ 2 :
   NEXT
```

¿os da el mismo incorrecto y asombroso resultado que a mí? ¿Por qué demonios?

4.—Dos truquitos basados en los "POKES" de control de la pantalla de trabajo.

A) **Temblor de tierra** (muy útil para acompañar a explosiones).

```
10 FORT = 1 TO 10 : POKE 36865,
   PEEK (36865)+1:FORI=1 TO 30:
   NEXTI:POKE 36865,PEEK(36865)—1
20 FORJ=1TO50:NEXTJ,T
```

.T: número de veces que se ejecuta cada pareja "subida-bajada".

.+1,-1: Altura del movimiento (intervalo. Modificable desde el ligero temblor a la sacudida).

.I: Tiempo muerto entre cada subida y cada bajada.

.J: Tiempo muerto entre cada bajada y cada subida.

B) **Efecto "paso de diapositiva"**

```
10 FORI=1 TO 35:POKE 36864,J:NEXT:
   FOR I=35 TO 1 STEP—1:POKE
   36864,J: NEXT:POKE 36864,12
```

.Sin comentarios. Aún es mejor sustituyendo la última orden por

```
FOR K=1 TO 12:POKE 36864,K:NEXT K.
```

GUILLERMO FATAS
STA. TERESA, 58-2º
50006 ZARAGOZA

1. 32 K no se pueden tener a la vez. El máximo son 28 K=16K+8K+3K de memoria standard. Los otros 3K se guardan para rutinas en código máquina y no son accesibles desde Basic.

2. Esto se da cuando hay un error en los datos del archivo. Es un mensaje del KERNAL.

3. Se debe a pequeñas operaciones que realiza el ordenador para manejar los datos.

LOAD EN MODO DE PROGRAMA



El motivo de la presente es el siguiente: Tengo un Commodore 64. A la hora de enlazar diversos programas de una aplicación utilizo el comando "load", en modo de programa de la siguiente manera:

Por ejemplo:

```
if a=Ø then load "pepito",8
```

El resultado es que se pone a cargar el programa "pepito", pero no lo carga del todo, sino parte del mismo, además de cargar otras cosas tales como: signos raros entre los que se encuentran las palabras del Basic todas juntas, con cambios del color.

El código máquina no lo empleo para nada a excepción de algún "poke" para el mapa de memoria de la pantalla. Tampoco toco ningún puntero de ninguna clase.

Esperando que haya quedado claro mi problema, se despide hasta otra ocasión,

ANDRES PASCUAL RINCON
FELIX LAFUENTE, 4-6º DCHA.
22004 HUESCA

La forma correcta de cargar el programa sería: POKE653,131:POKE198,1:END. Ahora bien, esto NO conserva las variables. Para conservarlas hay que hacer un LOAD en la forma normal, pero el primer programa ha de ser necesariamente MAYOR que el programa que se va a cargar.

¿INCOMPATIBILIDADES?



Me gustaría que me pudieran indicar acerca de este problema que tengo: Me compré un COMMODORE 64 y un datassette en Escocia y acabo de leer en la revista (COMMODORE WORLD, naturalmente) que puede haber y de hecho hay unas incompatibilidades entre los C-64 de Gran Bretaña y los de aquí, sobre todo al trabajar con ellos por pantalla; así que les estaría muy agradecido que me indicaran

...PREGUNTONES ¿ ?



cuáles son éstas diferencias y si son fáciles de modificar, y si tendré algún problema para acoplarle una impresora o cualquier otro periférico de aquí.

RAMON ARENAS RODRIGUEZ
CAMPOAMOR, 23
33001 OVIEDO

Creemos que no existen incompatibilidades entre los Commodore 64 de Escocia y los de España, la única que conocemos es que no recibirás el sonido por el altavoz de tu televisor, pero puedes solucionarlo conectando la salida trasera "audio output" de tu C-64 a la entrada de un amplificador de sonido exterior.

No sabemos de otro tipo de incompatibilidades con ningún tipo de periféricos, pero para comprobarlo es mejor que te pongas en contacto con los distribuidores exclusivos de Commodore en España, que como ya sabrás es Micro Electrónica y Control.

MANUALES Y DISCRIMINACION



Primero felicitarles por vuestra-nuestra revista, la cual me gusta mucho. Me llamo Pablo y tengo 14 años. Me gustaría saber si me podéis enviar una fotocopia del manual de SIMON'S BASIC, ya que lo compré en Inglaterra y no sé traducirlo todo.

Ahora una sugerencia: por qué no publican programas en Código Máquina y programas para el SIMONS BASIC, para el lenguaje FORTH, etc.

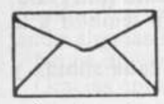
Me gustaría contactar con usuarios del COMMODORE 64 para intercambio de programas. Llamar de 10 a 11 o escribir a:

PABLO PERNAS GARCIA
GRAN VIA, 135 CHALET
VIGO (PONTEVEDRA)

En lo relativo al manual del Simon's Basic en castellano, tenemos que decirte que nosotros no publicamos este manual ni lo vendemos, además de que la única copia que tenemos del manual es el original en inglés, y no sabemos si Micro Electrónica lo habrá traducido al castellano.

Estamos intentando preparar artículos para los diferentes cartuchos: Simons, Bredens Basic, Vic-tree, Forth, File Boss, Master, etc., pero debes reconocer que son tantos y tan variados que es prácticamente imposible satisfacer a todos al mismo tiempo.

PROTECCIONES



Hola, quisiera saber si tenéis por ahí (o algún lectorcete), por lo que os agradecería que publicárais esta nota en "seamos preguntones" y "Market Club" algún programita para evitar que las grabaciones en cinta puedan ser copiadas sin autorización (para evitar "copias piratas" ya sabéis). A quien lo tenga, le agradecería que me lo comunicara.

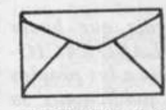
Lo necesito para el C-64. Otra cosa: ¿quién tiene por ahí a mano la correspondencia entre posiciones de memoria (POKE) del VIC-20 y el C-64?

JOSE LOPEZ
C/MORATIN, 5 - BUZON 5
28014 MADRID

En este mismo número hemos publicado en el cursillo de lenguaje máquina la forma en que hacía el auto-run el comando OLD, creo que es una buena forma de proteger programas. En Barcelona están preparando una serie sobre protecciones que cuando se termine esperamos poder publicar... claro que al publicar una protección se está dando automáticamente la forma de desprotegerlo, de modo que creo que la mejor protección que podías ponerle a tus programas es aquella que se te ocurriese a tí mismo y no se la dijeses a nadie.

Ten en cuenta que toda protección se puede levantar por complicada que sea, y siempre hay alguien capaz de hacerlo (yo conozco uno que ha levantado todo lo que ha cogido...). La mejor protección que pueden poner en sus productos las casas, es una buena información, manuales, precios competitivos, etc. que hagan preferir el producto original frente a las copias.

LIMPIEZA DE CABEZAS



Para limpiar el datassette, qué es preferible algodón y alcohol o una cinta limpiadora, yo les pregunto: cualquier cinta limpiadora o alguna especial.

Qué cartuchos de ayuda al programador existen para el VIC-20 a excepción del Superexpander y sus precios.

LUIS JORGE LOPEZ

En principio es indiferente, pero existen algunas diferencias: la cinta limpiadora no deja residuos, y el algodón impregnado en alcohol puede dejarlos pero un poco de algodón es más barato que una cinta limpiadora, si tienes cuidado con los restos de algodón que se quedan en la cabeza, el resultado es similar y puedes utilizar los dos métodos indistintamente.

Entre los cartuchos de ayuda al programador te puedo citar sólo el que tú mencionas, el de ayuda al programador y el vic-tree, pero los precios deberás preguntarlos a sus distribuidores respectivos ya que no los tengo (Micro Electrónica y Sakati).

VIDEO CASINO

Sopa de letras

Por David D. BUSCH (RUN. EE.UU.)

Traducido por Valerie SHANKS y adaptado por Alvaro IBAÑEZ

Te va a costar trabajo encontrar tu propio nombre en este laberinto.

En la sección de "Video Casino" hemos procurado publicar un surtido de programas, unos controlados por el joystick, otros estilo "marcianitos" y otros que son los tradicionales juegos de palabras adaptados al ordenador. Este tipo de juegos siempre ha sido bien aceptado por los usuarios del ordenador personal dado que son mucho más fáciles de escribir y jugar que los juegos que requieren gráficos complicados.

El juego "Sopa de Letras" es una variante de los juegos en que una palabra o serie de palabras se esconden en una matriz de letras. Para empezar hemos incluido unas cuantas palabras en este programa, pero puedes añadir palabras tuyas si quieres. No te preocupes si empiezas a familiarizarte con una de las palabras y la encuentras con facilidad, tendrás poblemas para encontrar tu propio nombre.

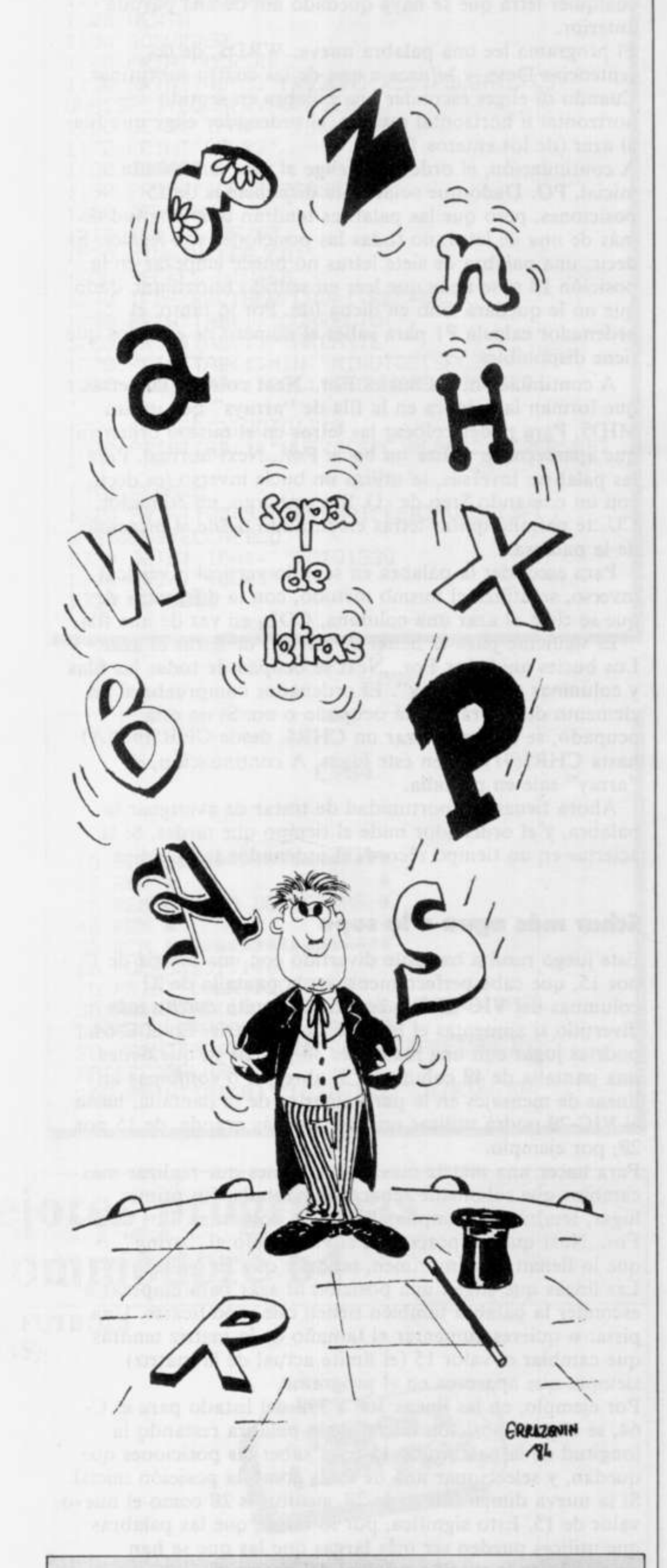
Escondiendo la letra

En este juego, sólo se oculta una palabra a la vez, dentro de una matriz de 15 letras por 15 letras. La palabra puede ser ocultada de forma horizontal o vertical, hacia adelante o hacia atrás, pero no en sentido diagonal. El ordenador te indica el tiempo que tardas en buscar la palabra.

Dieciocho palabras están incluidas en las líneas 200-240 del programa (líneas 200-245 en el listado del VIC-20). Puedes añadir todas las palabras que quieras. Sin embargo, por mucho que juegues, aunque sea con las mismas palabras, no te vas a aburrir dado que el ordenador elige al azar una nueva posición inicial cada vez que empieza y puede invertir el orden de las letras, creando una matriz diferente.

Dado que una matriz no es más que un "array" de dos dimensiones, las letras se almacenan en un "array", PZ$(15,15). Una vez que el ordenador haya colocado la palabra en el "array" las posiciones que quedan se llenan de letras elegidas al azar.

En cada partida el ordenador empieza seleccionando un



VIC-20 y C-64

número de 1 a 4. Este número, C, determina el sentido en que se va a ocultar la palabra: horizontal, horizontal inverso, vertical o vertical inverso. A continuación, el "array" existente recibe un valor nulo para eliminar cualquier letra que se haya quedado allí de una partida anterior.

El programa lee una palabra nueva, WRD$, de las sentencias Data, y bifurca a una de las cuatro subrutinas. Cuando tú eliges esconder una palabra en sentido horizontal u horizontal inverso, el ordenador elige una fila al azar (de los enteros 1 a 15).

A continuación, el ordenador elige al azar la posición inicial, PO. Dado que solamente disponemos de 15 posiciones, pero que las palabras tendrán una longitud de más de una de letra, no todas las posiciones son legales. Es decir, una palabra de siete letras no puede empezar en la posición 1Ø si se tiene que leer en sentido horizontal, dado que no le quedará sitio en dicha fila. Por lo tanto, el ordenador calcula P1 para saber el número de espacios que tiene disponibles.

A continuación, los bucles For...Next colocan las letras que forman la palabra en la fila de "arrays" que utiliza MID$. Para poder colocar las letras en el mismo orden en que aparecen, se utiliza un bucle For...Next normal. Para las palabras inversas, se utiliza un bucle inverso (es decir, con un comando Step de -1). Sin embargo, un contador, CU, te permite quitar letras empezando desde el principio de la palabra.

Para esconder la palabra en sentido vertical o vertical inverso, se utiliza el mismo método, con la diferencia de que se elige al azar una columna, COL, en vez de una fila.

El siguiente paso es llenar el "array" de letras al azar. Los bucles anidados For...Next se ocupan de todas las filas y columnas en el "array". El ordenador comprueba si un elemento del "array" está ocupado o no. Si no está ocupado, se coloca al azar un CHR$, desde CHR$(65)(A) hasta CHR$(91)(Z), en este lugar. A continuación, el "array" sale en pantalla.

Ahora tienes la oportunidad de tratar de averiguar la palabra, y el ordenador mide el tiempo que tardes. Si la aciertas en un tiempo récord, el ordenador te lo indica.

Echar más agua a la sopa

Este juego resulta bastante divertido con una matriz de 15 por 15, que cabe perfectamente en la pantalla de 22 columnas del VIC-2Ø. Sin embargo, resulta mucho más divertido si aumentas el tamaño de la matriz. En el C-64 podrías jugar con una matriz de 3Ø por 2Ø ya que tienes una pantalla de 4Ø columnas. Si abrevias o combinas las líneas de mensajes en la parte inferior de la pantalla, hasta el VIC-2Ø podrá utilizar una matriz más grande, de 15 por 2Ø, por ejemplo.

Para hacer una matriz más grande tienes que realizar más cambios que solamente rehacer la pantalla. En primer lugar, tendrás que ampliar PZ$ (fila, columna). Los bucles For...Next que proporcionan el valor nulo al "string", o que lo llenan o lo imprimen, tendrán que ampliarse.

Las líneas que eligen una posición al azar para empezar a esconder la palabra también tienen que modificarse. Una pista: si quieres aumentar el tamaño de la matriz tendrás que cambiar el valor 15 (el límite actual de la matriz) siempre que aparezca en el programa.

Por ejemplo, en las líneas 36Ø a 39Ø del listado para el C-64, se elige la posición inicial de la palabra restando la longitud de la palabra de 15 para saber las posiciones que quedan, y seleccionar una de éstas como la posición inicial. Si la nueva dimensión es de 2Ø, sustituyes 2Ø como el nuevo valor de 15. Esto significa, por lo tanto, que las palabras que utilices pueden ser más largas que las que se han empleado en la versión original del programa.

Asegúrate de que tengas una copia de reserva antes de empezar a hacer modificaciones por si te equivocas.

VIC-20

```
10 REM ***************
20 REM *             *
30 REM *   SOPA DE   *
40 REM *   LETRAS    *
45 REM *             *
50 REM ***************
70 PRINT"[CLR][2CRSRD]"
80 PRINTTAB(3)"[GRN][RVSON]SOPA[SPC]
DE[SPC]LETRAS[BLU][2CRSRD]"
90 PRINTTAB(1)"TE[SPC]MOSTRARE[SPC]U
NA"
100 PRINTTAB(1)"PALABRA,HAS[SPC]DE"
110 PRINTTAB(1)"ENCONTRARLA[SPC]EN[SPC]
EL"
120 PRINTTAB(1)"PUZZLE[SPC]TAN[SPC]R
APIDO"
125 PRINTTAB(1)"COMO[SPC]PUEDAS."
130 PRINTTAB(1)"INTRODUCE[SPC]LA[SPC]
FILA[SPC]Y"
135 PRINTTAB(1)"LA[SPC]COLUMNA[SPC]D
E[SPC]LA"
140 PRINTTAB(1)"PRIMERA[SPC]LETRA[SPC]
DE[SPC]LA[3SPC]PALABRA."
150 PRINTTAB(3)"[CRSRD][RVSON][RED]P
ULSA[SPC]UNA[SPC]TECLA[BLU]"
160 GETA$:IFA$=""THEN160
170 PRINT"[CLR]"
180 DIMPUZ$(15,15)
190 REC=999999999
200 DATA ORDENADOR,TECLADO
205 DATA TELEVISION,REVISTA
210 DATA COMMODORE,PROGRAMA
215 DATA CONECTOR,CORRIENTE
220 DATA PANTALLA,CHIP
225 DATA DISCO,IMPRESORA
230 DATA PLOTTER,JOYSTICK
240 DATA DIMENSION,ENCHUFE
245 DATA CINTA,FUNCION,#
250 C=INT(RND(1)*4)+1
260 FORR=1TO15
270 FORC1=1TO15
280 PUZ$(R,C1)=""
290 NEXTC1
300 NEXTR
310 READWRD$:IF WRD$="#" THEN END
320 PRINT"[CLR][SPC]"
330 PRINTWRD$
340 ON C GOTO 350,450,570,690
350 ROW=INT(RND(1)*15)+1
360 P=LEN(WRD$)
370 P1=15-P
380 PO=INT(RND(1)*P1):PO=3
390 C2=PO+1
400 R2=ROW
410 FORN=1TOLEN(WRD$)
420 PUZ$(ROW,N+PO)=MID$(WRD$,N,1)
430 NEXTN
440 GOTO790
450 ROW=INT(RND(1)*15)+1
460 P=LEN(WRD$)
470 P1=15-P
480 PO=INT(RND(1)*P1)
490 C2=PO+P
500 R2=ROW
510 FORN=PTO1STEP-1
520 CU=CU+1
530 PUZ$(ROW,N+PO)=MID$(WRD$,CU,1)
540 NEXTN
550 CU=0
560 GOTO790
570 COL=INT(RND(1)*15)+1
580 P=LEN(WRD$)
590 P1=15-P
```

```
600 PO=INT(RND(1)*P1)
610 R2=PO+P
620 C2=COL
630 FORN=PTO1STEP-1
640 CU=CU+1
650 PUZ$(N+PO,COL)=MID$(WRD$,CU,1)
660 NEXTN
670 CU=0
680 GOTO790
690 COL=INT(RND(1)*15)+1
700 P=LEN(WRD$)
710 P1=15-P
720 PO=INT(RND(1)*P1)
730 R2=PO+1
740 C2=COL
750 FORN=1TOP
760 PUZ$(N+PO,COL)=MID$(WRD$,N,1)
770 NEXTN
780 GOTO790
790 FORR=1TO15
800 FORC1=1TO15
810 B=INT(RND(1)*26)+65
820 IFPUZ$(R,C1)=""THENPUZ$(R,C1)=CH
R$(B)
830 NEXTC1
840 NEXTR
850 GOTO950
860 PRINT"[CLR][CRSRD]"
870 PRINTTAB(6)"[RED]ABCDEFGHIJKLMNO
[BLU]"
880 FORR=1TO15
890 PRINT"[RED]":PRINTTAB(2)R;TAB(6)
"[BLU]";
900 FORC1=1TO15
910 PRINTPUZ$(R,C1);
920 NEXTC1
930 NEXTR
940 RETURN
950 GOSUB860
960 PRINT"[CRSRD]"
970 PRINTTAB(1)"COLUMNA?[SPC](A-O):"

980 FI=TI
990 GETA$:IFA$=""GOTO990
1000 A=ASC(A$)
1010 IFA<65ORA>79GOTO990
1020 A=A-64
1030 SI=TI
1040 PRINTTAB(1)"";
1050 INPUT"FILA[SPC](1-15)";RT$
1060 RT=VAL(RT$)
1070 IFR2=RT AND A=C2 GOTO1190
1080 PRINT"[CLR][2CRSRD]"
1090 PRINTTAB(1)"TE[SPC]EQUIVOCASTE.
ESTABA"
1100 PRINTTAB(1)"EN[SPC]";CHR$(C2+64
);R2
1110 FORN=1TO2000
1120 NEXTN
1130 GOSUB860
1140 PRINT"[CRSRD]"
1150 PRINTTAB(3)"[RVSON][RED]PULSA[SPC]
UNA[SPC]TECLA[RVSOFF][BLU]"
1160 GETA$:IFA$=""THEN1160
1170 PRINT"[CLR]"
1180 GOTO250
1190 PRINT"[CLR][2CRSRD]"
1200 DF=SI-FI
1210 ES=INT(DF/60)
1220 MIN=INT(ES/60)
1230 SEC=ES-MIN*60
1240 IFDF<RECTHENFLAG=1:REC=DF
1250 PRINTTAB(1)"CORRECTO!!."
1260 PRINTTAB(1)"TARDASTE"
1270 PRINTTAB(1)MIN;"MINUTOS[SPC]Y"
1280 PRINTTAB(1)SEC;"SEGUNDOS.[3CRSRD]
"
1290 IFFLAG<>1GOTO1310
1300 PRINTTAB(1)"NUEVO[SPC]RECORD!![2CRSRD]
"
1310 FLAG=0
1320 PRINTTAB(3)"[RVSON][RED]PULSA[SPC]
UNA[SPC]TECLA[BLU]"
1330 GETA$:IFA$=""GOTO1330
1340 PRINT"[CLR]"
1350 GOTO250
```

C-64

```
10 REM ****************
20 REM *              *
30 REM * SOPA DE LETRAS *
40 REM *              *
50 REM ****************
60 POKE53281,140
70 PRINT"[CLR][2CRSRD]"
80 PRINTTAB(10)"[GRN][RVSON]SOPA[SPC]
DE[SPC]LETRAS[BLU][2CRSRD]"
90 PRINTTAB(3)"TE[SPC]MOSTRARE[SPC]U
NA[SPC]PALABRA."
100 PRINTTAB(8)"INTENTA[SPC]ENCONTRA
```

```
RLA"
110 PRINTTAB(8)"EN[SPC]EL[SPC]PUZZLE
[SPC]TAN[SPC]RAPIDO"
120 PRINTTAB(8)"COMO[SPC]PUEDAS."
130 PRINTTAB(8)"INTRODUCE[SPC]LA[SPC]
FILA[SPC]Y[SPC]LA"
140 PRINTTAB(8)"COLUMNA[SPC]DE[SPC]L
A[SPC]PRIMERA"
145 PRINTTAB(8)"LETRA[SPC]DE[SPC]LA[SPC]
PALABRA.[2CRSRD]"
150 PRINTTAB(10)"[RVSON][RED]PULSA[SPC]
UNA[SPC]TECLA[BLU]"
160 GETA$:IFA$=""THEN160
170 PRINT"[CLR]"
180 DIMPUZ$(15,15)
190 REC=999999999
200 DATA ORDENADOR,TECLADO
205 DATA TELEVISION,MEMORIA
210 DATA COMMODORE,PROGRAMA
215 DATA FUNCION,REVISTA
220 DATA MONITOR,VARIABLE
225 DATA DISCO,IMPRESORA
230 DATA GRAFICO,PANTALLA
235 DATA CONECTOR,CHIP
240 DATA ENCHUFE,DIMENSION,#
250 C=INT(RND(1)*4)+1
260 FORR=1TO15
270 FORC1=1TO15
280 PUZ$(R,C1)=""
290 NEXTC1
300 NEXTR
310 READWRD$:IFWRD$="#"THEN END
320 PRINT"[CLR][SPC]"
330 PRINTWRD$
340 ON C GOTO 350,450,570,690
350 ROW=INT(RND(1)*15)+1
360 P=LEN(WRD$)
370 P1=15-P
380 PO=INT(RND(1)*P1):PO=3
390 C2=PO+1
400 R2=ROW
410 FORN=1TOLEN(WRD$)
420 PUZ$(ROW,N+PO)=MID$(WRD$,N,1)
430 NEXTN
440 GOTO790
450 ROW=INT(RND(1)*15)+1
460 P=LEN(WRD$)
470 P1=15-P
480 PO=INT(RND(1)*P1)
490 C2=PO+P
500 R2=ROW
510 FORN=PTO1STEP-1
520 CU=CU+1
530 PUZ$(ROW,N+PO)=MID$(WRD$,CU,1)
540 NEXTN
550 CU=0
560 GOTO790
570 COL=INT(RND(1)*15)+1
580 P=LEN(WRD$)
590 P1=15-P
600 PO=INT(RND(1)*P1)
610 R2=PO+P
620 C2=COL
630 FORN=PTO1STEP-1
640 CU=CU+1
650 PUZ$(N+PO,COL)=MID$(WRD$,CU,1)
660 NEXTN
670 CU=0
680 GOTO790
690 COL=INT(RND(1)*15)+1
700 P=LEN(WRD$)
710 P1=15-P
720 PO=INT(RND(1)*P1)
730 R2=PO+1
740 C2=COL
750 FORN=1TOP
760 PUZ$(N+PO,COL)=MID$(WRD$,N,1)
770 NEXTN
780 GOTO790
790 FORR=1TO15
800 FORC1=1TO15
810 B=INT(RND(1)*26)+65
820 IFPUZ$(R,C1)=""THENPUZ$(R,C1)=CH
R$(B)
830 NEXTC1
840 NEXTR
850 GOTO950
860 PRINT"[CLR][CRSRD]"
870 PRINTTAB(12)"[RED]ABCDEFGHIJKLMN
O[BLU]"
880 FORR=1TO15
890 PRINT"[RED]":PRINTTAB(8)R;TAB(12
)"[BLU]";
900 FORC1=1TO15
910 PRINTPUZ$(R,C1);
920 NEXTC1
930 NEXTR
940 RETURN
950 GOSUB860
960 PRINT"[CRSRD]"
970 PRINTTAB(8)"COLUMNA[SPC]?[SPC](A
-O):"
980 FI=TI
990 GETA$:IFA$=""GOTO990
1000 A=ASC(A$)
1010 IFA<65ORA>79GOTO990
1020 A=A-64
1030 SI=TI
1040 PRINTTAB(8)"";
1050 INPUT"FILA[SPC](1-15)";RT$
1060 RT=VAL(RT$)
1070 IFR2=RT AND A=C2 GOTO1190
1080 PRINT"[CLR][2CRSRD]"
1090 PRINTTAB(8)"TE[SPC]EQUIVOCASTE.
"
1100 PRINTTAB(8)"ESTABA[SPC]EN[2SPC]
";CHR$(C2+64);"-";R2
1110 FORN=1TO2000
1120 NEXTN
1130 GOSUB860
1140 PRINT"[CRSRD]"
1150 PRINTTAB(10)"[RVSON][RED]PULSA[SPC]
UNA[SPC]TECLA[RVSOFF][BLU]"
1160 GETA$:IFA$=""THEN1160
1170 PRINT"[CLR]"
1180 GOTO250
1190 PRINT"[CLR][2CRSRD]"
1200 DF=SI-FI
1210 ES=INT(DF/60)
1220 MIN=INT(ES/60)
1230 SEC=ES-MIN*60
1240 IFDF<RECTHENFLAG=1:REC=DF
1250 PRINTTAB(8)"CORRECTO!."
1260 PRINTTAB(8)"TARDASTE[SPC]EXACTA
MENTE"
1270 PRINTTAB(8)MIN;"MINUTOS[SPC]Y"
1280 PRINTTAB(8)SEC;"SEGUNDOS.[3CRSRD]
"
1290 IFFLAG<>1GOTO1310
1300 PRINTTAB(8)"NUEVO[SPC]RECORD!!
[2CRSRD]"
1310 FLAG=0
1320 PRINTTAB(10)"[RVSON][RED]PULSA[SPC]
UNA[SPC]TECLA[BLU]"
1330 GETA$:IFA$=""GOTO1330
1340 PRINT"[CLR]"
1350 GOTO250
```

El Misterio de los Pokes (VIII)



Por Diego ROMERO

En primer lugar debemos pedir disculpas por el despiste del mes pasado, en el que por error se omitieron los listados del programa de ayuda gráfica del artículo titulado "Pokes sin dolor".

Después de deciros que hemos preparado una tabla con las direcciones de llamadas de las distintas rutinas del kernal que se deben modificar para adaptar los programas del VIC-2Ø al C-64 y viceversa. Esta tabla puede resultaros útil para descifrar algo del modo en que opera vuestro ordenador. En la tabla mencionamos la dirección de llamada del C-64, la del VIC-2Ø y la descripción de su función.

C-64	VIC-2Ø	Descripción
E45F	E429	Mensajes del sistema operativo.
E5ØØ	E5ØØ	Devuelve la dirección de la CIA o VIA.
E5Ø5	E5Ø5	Devuelve el formato de pantalla.
E5ØA	E5ØA	Fija o lee la posición del cursor.
E544	E55F	Borra la pantalla.
E5CA	E5E5	Espera una entrada del teclado.
E632	E64F	Lee un dato de la pantalla.
E8EA	E975	Desplaza la pantalla (Scroll).
E9FF	EA8D	Borra una línea de pantalla.
EA31	EABF	Rutina de interrupción (IRQ).
EDØ9	EE14	Envía TALK a un periférico.
EDØC	EE17	Envía LISTEN.
ED4Ø	EEE4	Envía un byte al bus serie.
EDB9	EECØ	Envía dirección secundaria para LISTEN.
EDC7	EECE	Envía dirección sec. para TALK.
EDEF	EEF6	Envía UNTALK.
EDFE	EFØ4	Envía UNLISTEN.
EE13	EF19	Lee un dato del bus serie.
EEBB	EFA3	Salida de RS232.
FØ14	FØED	Salida al buffer de RS 232.
FØ86	F14F	Entrada del RS 232.
FØBD	F174	Mensajes de error del sistema.
F157	F2ØE	BASIN lee un dato (get).
F1CA	F27A	BSOUT escribe un carácter.
F2ØE	F2C7	CHKIN fija dispositivo de entrada.
F25Ø	F3Ø9	CHKOUT idem de salida.
F291	F34A	CLOSE.
F3ØF	F3CF	SETNAM fija nombre de OPEN.
F31F	F3DF	SETLFS fija parámetros de OPEN.
F32F	F3EF	CLALL cierra todos los canales.
F34A	F4ØA	OPEN
F49E	F542	LOAD
F5AF	F647	Escribe "SEARCHING FOR...".
F5D2	F66A	Escribe LOADING o VERIFYING.
F5DD	F675	SAVE.
F68F	F728	Escribe "SAVING...".
F69B	F734	UDTIM actualiza el reloj.
F6DD	F76Ø	Lee hora.
F6E4	F767	Escribe hora.
F72C	F7AF	Lee la cabecera de un programa de cinta.
F76A	F7E7	Escribe la cabecera en cinta.
F7EA	F867	Comprueba el nombre.
F841	F8CØ	Lee de cinta.
F864	F8EA	Escribe a cinta.
FCCA	FDØ8	Para el motor del cassette.
FCE2	FD22	RESET del ordenador.
FDØ2	FD3F	Comprueba si tiene cartucho ROM.
FD1Ø	FD4D	Secuencia de autostart para ROM.
FEØ7	FE57	Comprueba el estado.
FE25	FE73	Fija o lee el límite superior del RAM.
FE34	FE82	Fija o lee el límite inferior del RAM.
FE43	FEA9	Rutina NMI.
FEC2	FF5C	Constantes de tiempo para RS 232.
FF81	FF8A	Tabla de saltos de las rutinas Kernal.

También puede servir de ayuda para la conversión de programas el decir que EN GENERAL las rutinas del sistema operativo inferiores a $EØØØ, para calcular la dirección del VIC conociendo la del C-64 debemos sumarle 8192 ($2ØØØ), y a partir de la $EØØØ hasta $E5ØØ debe restarse 3 para conseguir la dirección que correspondería a VIC-2Ø. Esta regla es general y podemos encontrar rutinas que no cumplan esta regla.

Espero que las tablas y normas sirvan para poder adaptar los programas de código máquina de uno a otro ordenador, teniendo en cuenta las limitaciones de memoria, pantalla y gráficos del VIC frente a su hermano mayor el 64. También creo que será útil el programa de ayuda gráfica como ya dije en el número anterior.

Listados del artículo POKE SIN DOLOR del nº anterior

VIC-20

```
10 REM ***************
15 REM
20 REM AYUDA AL DIBUJO
30 REM    DE GRAFICOS
40 REM
45 REM VERSION VIC-20
50 REM
55 REM      POR
56 REM
60 REM CRAIG HOWARTH
90 REM
95 REM **************
96 REM   INICIALIZA
97 REM    VARIABLES
98 REM **************
99 REM
100 IFPEEK(44)=18THEN105
103 SS=7680:CS=38400:GOTO107
105 SS=4096:CS=37888
107 CD=CS-SS
110 SC=36879:JY=37137:JK=37154
120 TL=SS+22:BL=SS+505:PO=SS+252:PC=
PO+CD
130 DIMJO(13)
140 FORI=1TO13:READJO(I):NEXT
150 DATA21,-23,-1,,22,-22,,,,1,-21,
23
155 SP$=CHR$(19):FORI=1TO22:SP$=SP$+
CHR$(32):NEXT:SP$=SP$+CHR$(8)
200 REM
201 REM *******
202 REM TITULOS
203 REM *******
204 REM
205 PRINTSP$
210 PRINT"[HOM][2SPC]AYUDA[SPC]AL[SPC]
DIBUJO[4SPC][HOM]":FORT=1TO1000:NEXT

212 PRINT"[HOM][4SPC]DE[SPC]GRAFICOS
[3SPC][HOM]":FORT=1TO1000:NEXT
215 PRINT"[HOM][2SPC]POR[SPC]CRAIG[SPC]
HOWART[3SPC][HOM]":FORT=1TO1000:NEXT

220 PRINT"[HOM][SPC][RVSON]P[RVSOFF]
ANTALLA[SPC]O[SPC][RVSON]I[RVSOFF]MP
RESORA[HOM]"
230 A$="":GETA$:IFA$=""THEN230
240 IFA$="I"THEN1000
245 IFA$="P"THEN300
250 GOTO230
300 REM
301 REM ***********
302 REM  SALIDA A
303 REM  PANTALLA
304 REM ***********
305 REM
310 CN=PEEK(PO):CO=PEEK(PC)AND15
320 POKEPC,3:POKEPO,42
325 PRINTSP$
330 PRINTCHR$(19);PO;CN;PC;CO;CHR$(1
9)
340 POKEJK,127:JS=(PEEK(JY)AND28)OR(
PEEK(JY+15)AND128):POKEJK,255
342 JS=ABS(JS-100)/4-7
345 TP=PO+JO(JS)
350 IFTP<TLORTP>BLORJS=7THEN340
360 POKEPC,CO:POKEPO,CN
370 PO=TP:PC=PC+JO(JS)
380 GOTO310
1000 REM
1001 REM ***********
1002 REM   SALIDA A
1003 REM  IMPRESORA
1004 REM ***********
1005 REM
1010 PRINTSP$
1020 OPEN4,4:CMD4
1030 PRINTCHR$(16)"24AYUDA[SPC]AL[SPC]
DIBUJO[SPC]DE[SPC]GRAFICOS"
1040 PRINTCHR$(16)"31POR[SPC]CRAIG[SPC]
HOWART"
1050 PRINT:PRINT
1060 PRINT"COLOR[SPC]DEL[SPC]BORDE[SPC]
Y[SPC]DE[SPC]LA[SPC]PANTALLA:";PEEK(
SC)
1080 PRINT
1090 PRINT"[4SPC]PANTALLA",
1100 PRINTCHR$(16)"20OFFSET";
1110 PRINTCHR$(16)"32CODIGO[SPC]CARA
CTER";
1120 PRINTCHR$(16)"53COLOR";
1130 PRINTCHR$(16)"66CODIGO[SPC]COLO
R"
1140 PR=TL
1150 CN=PEEK(PR):CO=PEEK(PR+CD)AND15

1153 POKEPR+CD,0:POKEPR,42
1155 IFCN=32ORCN=96THEN1205
1160 PRINTCHR$(16)"05";PR;
1170 PRINTCHR$(16)"21";PR-SS;
1180 PRINTCHR$(16)"37";CN;
1190 PRINTCHR$(16)"53";PR+CD;
1200 PRINTCHR$(16)"70";CO
1205 POKEPR+CD,CO:POKEPR,CN
1210 PR=PR+1
1220 IFPR>BLTHEN1999
1230 GOTO1150
1999 PRINT#4:CLOSE4
```

C-64

```
10 REM ******************************
**
20 REM *
*
30 REM * AYUDA AL DIBUJO DE GRAFICOS
*
40 REM *
*
60 REM *        VERSION C-64
*
70 REM *
*
75 REM *      POR CRAIG HOWART
*
78 REM *
*
80 REM ******************************
**
90 REM
95 REM ********************
96 REM INICIALIZA VARIABLES
97 REM ********************
98 REM
100 SS=1024:CS=55296:CD=CS-SS
110 BC=53280:SC=53281:JY=56320
120 TL=1064:BL=2023:PO=SS+500:PC=PO+
CD
130 DIMJO(10)
140 FORI=1TO10:READJO(I):NEXT
150 DATA-40,40,,-1,-41,39,,1,-39,41
155 SP$=CHR$(19):FORI=1TO40:SP$=SP$+
```

```
CHR$(32):NEXT:SP$=SP$+CHR$(19)
200 REM
201 REM *******
202 REM TITULOS
203 REM *******
204 REM
205 PRINTSP$
210 PRINT"[HOM][2SPC]AYUDA[SPC]AL[SPC]
DIBUJO[SPC]DE[SPC]GRAFICOS[9SPC][HOM]
":FORT=1TO1000:NEXT
215 PRINT"[HOM][10SPC]POR[SPC]CRAIG[SPC]
HOWART[11SPC][HOM]":FORT=1TO1000:NEX
T
220 PRINT"[HOM]ELIJA[SPC]SALIDA:[SPC]
[RVSON]P[RVSOFF]ANTALLA[SPC]O[SPC][RVSON]
I[RVSOFF]IMPRESORA[HOM]"
230 GETA$:IFA$=""THEN230
240 IFA$="I"THEN1000
245 IFA$="P"THEN300
250 GOTO230
300 REM
301 REM *****************
302 REM SALIDA A PANTALLA
303 REM *****************
304 REM
310 CN=PEEK(PO):CO=PEEK(PC)AND15
320 POKEPC,1:POKEPO,42
325 PRINTSP$
330 PRINTCHR$(19);PO;PO-SS,CN;PC;CO;
CHR$(19)
340 JS=15-(PEEK(JY)AND15)
345 TP=PO+JO(JS)
350 IFTP<TLORTP>BLORJS=0THEN340
360 POKEPC,CO:POKEPO,CN
370 PO=TP:PC=PC+JO(JS)
380 GOTO310
1000 REM
```

```
1001 REM *****************
1002 REM SALIDA A IMPRESORA
1003 REM *****************
1004 REM
1010 PRINTSP$
1020 OPEN4,4:CMD4
1030 PRINTCHR$(16)"26AYUDA[SPC]AL[SPC]
DIBUJO[SPC]DE[SPC]GRAFICOS"
1040 PRINTCHR$(16)"31POR[SPC]CRAIG[SPC]
HOWART"
1050 PRINT:PRINT
1060 PRINT"COLOR[SPC]DEL[SPC]BORDE:"
;PEEK(BC)AND15
1070 PRINT"COLOR[SPC]DE[SPC]LA[SPC]P
ANTALLA:";PEEK(SC)AND15
1080 PRINT
1090 PRINT"[4SPC]PANTALLA";
1100 PRINTCHR$(16)"20OFFSET";
1110 PRINTCHR$(16)"32CODIGO[SPC]CARA
CTER";
1120 PRINTCHR$(16)"54COLOR";
1130 PRINTCHR$(16)"66CODIGO[SPC]COLO
R"
1140 PR=TL
1150 CN=PEEK(PR):CO=PEEK(PR+CD)AND15

1153 POKEPR+CD,1:POKEPR,42
1155 IFCN=32ORCN=96THEN1205
1160 PRINTCHR$(16)"05";PR;
1170 PRINTCHR$(16)"21";PR-SS;
1180 PRINTCHR$(16)"37";CN;
1190 PRINTCHR$(16)"53";PR+CD;
1200 PRINTCHR$(16)"70";CO
1205 POKEPR+CD,CO:POKEPR,CN
1210 PR=PR+1
1220 IFPR>BLTHEN1999
1230 GOTO1150
1999 PRINT#4:CLOSE4
```

(Viene de pág. 51.)

```
[SHIFT@][SPC][SHIFTL][SHIFT@][SPC][COMMG]
[BLK][2COMM+]",
227 PRINT"[RVSON][COMM+][YEL][COMMM]
[7SPC][COMMG][SPC][COMMM][6SPC][SPC]
[COMMG][BLK][2COMM+]";
228 PRINT"[RVSON][COMM+][YEL][COMMM]
[7SPC][COMMG][SPC][COMMM][7SPC][COMMG]
[BLK][2COMM+]";
229 PRINT"[RVSON][GRN][21COMMU]"
300 X=0:N=0:R=8110:R1=R+30720:GOSUB1
500
400 IFPEEK(197)=39THENGOSUB1000:Z=-2
2:GOSUB1500
410 IFPEEK(197)=63THENGOSUB1000:Z=22
:GOSUB1500
420 IFPEEK(197)=17THENGOSUB1000:Z=-1
:GOSUB1500
430 IFPEEK(197)=18THENGOSUB1000:Z=1:
GOSUB1500
520 READY:X=X+Y:A1=8100+X:A2=A1+3072
0
530 IFY=9THEN3300
580 IFPEEK(A1)=28THENPOKEA1,30:POKEA
2,3:GOTO3000
590 IFPEEK(A1)=0THENPOKEA1,30:POKEA2
,5:GOSUB2000:GOTO400
600 POKEA1,30:POKEA2,5
610 FORI=1TO100:NEXT:POKEA1,32:GOTO4
00
1000 POKER,0:R1=R+30720:POKER1,6:RET
URN
1500 R=R+Z:N=N+1
```

```
1510 IFPEEK(R)=230ORPEEK(R)=204ORPEE
K(R)=207THENR=R-Z
1520 IFPEEK(R)=227ORPEEK(R)=228ORPEE
K(R)=229THENR=R-Z
1530 IFPEEK(R)=208ORPEEK(R)=231ORPEE
K(R)=250ORPEEK(R)=248THENR=R-Z
1540 IFPEEK(R)=0THENN=N-1
1545 IFN=118THEN3100
1550 POKER,28:R1=R+30720:POKER1,1:RE
TURN
2000 FORI=1TO100:NEXT:POKEA1,0:POKEA
2,6:RETURN
3000 PRINT"[CLR][10CRSRD][7CRSRR][WHT]
PICADO!!!":GOTO3500
3100 POKE36869,240:POKE36879,27:PRIN
T"[CLR][3CRSRD][7CRSRR][RED][RVSON]B
RAVO!!!"
3110 PRINT"[6CRSRD][2CRSRR][BLK]TIEN
ES"N*10"PUNTOS[BLU]":PRINT"[7CRSRD]"
:END
3300 PRINT"[CLR][4CRSRD][CRSRR][RVSON]
[WHT]SE[SPC]ACABO[SPC]EL[SPC]TIEMPO!

3500 PRINT"[6CRSRD][2CRSRR][WHT]TENI
AS"N*10"PUNTOS":POKE36869,240:RUN10
5000 DATA1,1,1,-22,-22,-22,-22,-22,-
22,-22,-22,-22,-22,-22
5010 DATA22,-1,-1,-1,22,22,22,22,22,
1,1,1,1,1,1,1,1,1,1
5020 DATA22,22,22,22,1,1,1,-22,-22,-
22,-22
5030 DATA1,1,1,-22,-22,-22,-22,-22,-
22,22,-1,-1,-1,-1,-1,-1
```

(Pasa a pág. 71.)


Las Aventuras de KBIT KBITA RAMK Y ROMO
FELIZ NAVIDAD
por Errazquin 84


SNIFF
¿?
COMMODORE WORLD


PERO.. ¿SE PUEDE SABER QUE OS PASA, QUE LLEVAIS ASI TODO EL DIA?
SNIFFF


NADA.... SNIFF GLUP... SNIFFF
38
COMMODORE WORLD


UN MOMENTO... A MI DE SECRETITOS NADA... YO OS HE TRAIDO AQUI, Y SOY VUESTRO AMIGO.. ASI QUE...
¡!


.. PUES VERAS... ESTABAMOS LEYENDO COMMODORE WORLD... Y NOS HEMOS DADO CUENTA QUE HOY ES NAVIDAD..
NAVIDAD
?


..Y AQUI DENTRO NO TOMAREMOS TURRON... NI HABRÁ REGALOS.. NI FIESTA... NI NADA... GLUP! ¡AUUUUUUU... BUAAAAA... SNIF!


¡CARAMBA!... ES VERDAD.. AQUI NO HAY NAVIDAD... HUMMM... HABRÁ QUE PENSAR EN ALGO PARA CELEBRARLA
AJÁ... HE DE AVISAR AL EMPERADOR..




Y VA A CONTARLE EL CHISME AL EMPERADOR... LE VOY A...


POK
ZIUU
ZUOOSS
¡TOMA EN TODO EL COCO!
ERRAZQUIN/84


COMMODORE WORLD
RRRRRRRR
PLAF

¿SE PUEDE SABER QUE TE PASA ... ? .. ¿EH?
COMMODORE WORLD

... Y ELLOS DIJERON QUE ... BLA, BLA . BLA .. Y CELEBRAR ... BLA, BLA, BLA

A MI LA GUARDIAAAAA

ENTRE TANTO ....
VAMOS A DAR UNA VUELTA A VER SI SE OS PASA EL CABREO..

... Y EN EL CASTILLO ...
... Y NO VOLVAIS SI NO CUMPLIS MIS ORDENES

... Y POR FIN ...
..TU.. EL DEL CASCABEL .. QUEDAS DETENIDO POR ORDEN DEL EMPERADOR
¿?

MIRA POR DONDE SE VAN A LLEVAR ESTOS ENCIMA LA FIESTA DE NAVIDAD ...
NOOO ... QUIETO .. SON MUCHOS SOLDADOS ...

IRÉ CON ELLOS ... ES MEJOR .. AVISAD A LOS DEMÁS Y PONEROS EN MARCHA ... ¡RÁPIDO!
¿Y NO LES VAMOS A SACUDIR?

AVISAREMOS PRIMERO A HENRI ... CORREEEEEE
NO ENTIENDO NADA

AYYYYY... MIRA.. TAMBIEN SE LO HAN LLEVADO
SOCORROO

SE HAN LLEVADO A TODOS... SOLO QUEDO YO..
ERRAZQUIN/84


... PUES SI ESTAMOS SOLOS .. HEMOS DE SALVARLOS SIN AYUDA DE NADIE... ¡VAMOS!


PERO .. QUE ... VOTO A ..
AL PALACIO
ZUUUMMM


... ESO ES... VAN HACIA PALACIO A TODA VELOCIDAD...
PERFECTO.... LOS RECIBIREMOS COMO SE MERECEN.. JI, JI, JI
AL PALACI


AQUI ESTAMOS... ESTO ES MUY RARO NO HAY GUARDIAS EN LA PUERTA.. ... ES IGUAL.. ENTREMOS..
CASTILLO DEL EMPERADOR
VISITAS DE 5 A 8


VAMOS A LAS MAZMORRAS.. SEGURO QUE LOS TIENE ALLÍ
MAZMORRA


¡VACIA!
HOLA


SEGURO ENTONCES QUE LOS TIENE EN EL SALON DEL TRONO...
ZUOOSSSS


PASA TU PRIMERO


ES MEJOR QUE PASEIS VOSOTROS Y VEAIS LO QUE HA HECHO EL EMPERADOR
¿?




FELIZ NAVIDAD KBIT y KBITA
EL EMPE HA PREPARADO ESTA FIESTA PARA VOSOTROS PORQUE ES NAVIDAD


GRACIAS A EMPE VAMOS A TENER FIESTA ..¡QUE TODOS VOSOTROS AMIGOS LECTORES TENGAIS TAMBIEN UNA FELIZ NAVIDAD Y PRÓSPEÑO AÑO 1985!
Feliz Navidad
ERRAZQUIN

DIRECTORIO



DISTRIBUIDORES DE:

ORIC-1
CASIO FP200
ROCKWELL-AIM-65
VIDEO GENIE-EG-2000
CASIO FX-9000P
SINCLAIR SPECTRUM
OSBORNE 1
DRAGON-32
NEW BRAIN
EPSON HX-20

C/SANDOVAL, 3, 4, 6 MADRID 28010
TELEFONOS 445 18 33/18 70
TELEX 47784 SAVL-E

---

casa de software s.a.

casa de software, s.a.
c/ aragón, 272, 8º, 6.ª
tel. 215 69 52
08007 barcelona

REM

• Ordenadores personales Hard y Soft.
• Cursos de Basic.

Oficinas: **RENOVACION EN MARCHA, S.A.**
c/. Espronceda, 34 - 2º int. - MADRID-3
Teléfono **(91) 441 24 78**
Tienda: **REM SHOP 1**
c/. Galileo, 4 - MADRID-15
Teléfono **(91) 445 28 08**

B.M.

AVD. CESAR AUGUSTO, 72
Tels.: 23 56 82 y 22 65 44
50003—ZARAGOZA

CENTRO DE INFORMATICA

Las Rozas - Majadahonda
EMPEZAMOS
Cursillos en BASIC
cada 15 días
Directamente en ordenadores
VIC-20, COMMODORE 64
SPECTRAVIDEO
Teléfono: 637 31 51

FERRE-MORET S.A.

C/ Tusset, 8 - Entresuelo 2.ª
Tel.: 218 02 93
08006 BARCELONA

RADIO WATT

- ORDENADORES PERSONALES
- ACCESORIOS INFORMATICA
- COMPONENTES ELECTRONICOS
- TELECOMUNICACIONES

Paseo de Gracia 126-130
Tel. 237 11 82. 08008 BARCELONA*

MICROS GARDEN S.A.
- ORDENADORES PERSONALES -

- SINCLAIR SPECTRUM Y PLUS
- COMMODORE 64
- ATARI 800 XL
- ORIC ATMOS
- DRAGON 32 y 64

- PERIFERICOS Y ACCESORIOS
- SOFTWARE PARA TODAS LAS MARCAS
- CURSOS DE BASIC PARA PRINCIPIANTES (PRACTICAS CON ORDENADOR)
- GRAN SURTIDO EN LIBROS Y REVISTAS

Francisco Silvela, 19
Teléfono 401 07 27. 28028 MADRID

MicroSistemas

Todos los meses publicaremos programas para los Commodore. ¡¡Suscríbete!!

Tel. (91) 231 23 88/95

COMPUTERWORLD/ESPAÑA

es miembro de CW Communications/Inc., el grupo editorial más grande y de publicaciones relacionadas con los ordenadores en 18 países, dando como resultado la cantidad de nueve millones de lectores que leen una o más de nuestras publicaciones.

¡¡¡Suscríbete!!!
Tel. (91) 231 23 88/95

Este módulo de 4 cm. vale 55.000 Ptas. por 11 inserciones.
Llamarnos al 231 23 88/95

ORDENADORES
PERSONALES/PROFESIONALES

RED DE ORDENADORES PARA COLEGIOS.



MICROINFORMATICA

Equipos/Servicios
Centro de formación

General Ricardos, 15.
Tel. (974) 31 14 39. BARBASTRO (Huesca)

EXCURSION POR LA MICROPROGRAMACION

Cursillo de Lenguaje Máquina —VIII—

Por Diego ROMERO



Antes de proseguir con los ejemplos prácticos creo que es conveniente estudiar otra de las peculiaridades de las CPUs de nuestros ordenadores, son las interrupciones, que siendo una de las cosas más importantes en la programación en lenguaje-código máquina suelen ser omitidas o solamente mencionadas sin más detalles en casi todos los libros y manuales sobre el tema. Incluso algunos libros que son una magnífica fuente de información detallada del C-64 (así considero yo al "The Anatomy of the Commodore 64" de Abacus Software) en la sección sobre las CIAs del C-64 dice (en inglés) que si deseas utilizar las CIAs del C-64 para tus propios propósitos, debes fijarte que fueron diseñadas para funciones especiales en el computador, y esto es aplicable en particular al uso de las interrupciones... de modo que evite alterar el contenido del ICR (Registro de Control de Interrupciones).

Para entender las interrupciones mejor no puedo encontrar un simil en basic que se adapte perfectamente, pero tenemos uno en la vida cotidiana que nos viene como anillo al dedo:

Imaginemos por un momento que nos encontramos de noche en nuestra casa (que sería el ordenador), nosotros seríamos la cpu, y estamos realizando una tarea (ejecutando el programa) que podría ser leer el periódico en nuestro sillón favorito. En este momento suena el teléfono, y nosotros que tenemos que levantarnos a contestar, dejamos el periódico doblado por la página donde nos encontrábamos, contestamos a la llamada y volvemos al sillón para continuar nuestra apacible lectura. En este caso la llamada telefónica sería la "interrupción" con la que la cpu detuvo la ejecución del programa y luego la continuó. Este tipo de interrupción sería del tipo evitable, ya que si no queríamos recibir llamadas telefónicas habría sido suficiente con dejar descolgado el teléfono, o tener unos buenos cascos de música puestos en la cabeza para no oír la llamada.

Pero existe otro tipo de interrupciones que son las no evitables, y como su propio nombre indica no podemos ignorarlas, sería por ejemplo, un apagón de luz. En este caso no podríamos seguir nuestra lectura, deberíamos dejar el periódico doblado por el sitio donde nos encontrábamos y levantarnos a buscar una vela o intentar accionar el disyuntor de entrada de nuestra casa si este hubiese saltado.

Como veréis he procurado encontrar ejemplos muy sencillos, y ahora daré los términos algo más técnicos:

La CPU de nuestro sistema tiene dos líneas de interrupción (o más si consideramos la de reset...), que responden a las denominaciones de IRQ (Interrupt ReQuest) y NMI (Non Maskable Interrupt), por estas líneas puede recibir una señal de interrupción de cualquier dispositivo que precise su servicio (reloj, bus serie, teclado, etc.) y le hace interrumpir la ejecución del programa para atender su petición. Las primeras son las evitables, y el modo de impedir que interrumpan el programa es activar la bandera (flag) correspondiente del registro de estado de la cpu, y desactivarla cuando terminemos de ejecutar la parte importante del programa para permitir que la cpu pueda atender a las otras tareas. Pero la línea de NMI la utilizan aquellos dispositivos que no pueden esperar por ser más importantes que el programa principal, esta línea origina siempre la interrupción aunque tengamos activada la bandera del registro de estado de la cpu (en nuestro simil no serviría de nada haber dejado descolgado del teléfono si saltase el disyuntor de casa y nos quedásemos sin luz para leer).

Cuando nos interrumpían, dejábamos doblado el periódico para poder volver al punto donde habíamos dejado la lectura, esto también lo hace la cpu cuando se la interrumpe, coloca los registros internos en el stack (pila de datos), para cuando termine la rutina de interrupción (con la instrucción RTI) poder reasumir la ejecución del programa en el punto donde se encontraba.

Nuestro equipo se ha diseñado de modo que un temporizador de una de las VIA (en el VIC) o CIA (en el C-64), produzca 60 interrupciones por segundo del tipo evitable, de modo que atendamos a una serie de funciones internas 60 veces cada segundo como

EXCURSION POR LA MICROPROGRAMACION

puede ser el poner en hora el reloj interno (TI$), comprobar el teclado, etc. Alguno se preguntará para qué interrumpir a la cpu para "esas cosas", pues bien: en primer lugar la cpu lee bastante más rápido que nosotros lo hacemos con el periódico, y aunque la detengamos 60 veces por segundo no supone una gran pérdida de tiempo para ella. Por otro lado, si no lo hiciésemos por medio de interrupciones, tendríamos que hacerlo desviando el programa de vez en cuando a comprobar el teclado, etc., que sería equivalente a estar levantándonos cada medio minuto de nuestro asiento a descolgar el teléfono, decir ¡diga!, y volver a colgar si lo que oímos es el tono de marcar, y si lo hiciésemos cada media hora, puede que alguien con una llamada importante hubiese estado esperando cinco minutos y se hubiera cansado de esperar. Por lo tanto es mejor descolgar el teléfono sólo cuando suene (atender a los periféricos sólo cuando lo soliciten con IRQ), y cuando estemos haciendo algo importante dejarlo descolgado para no recibir llamadas (deshabilitar las IRQs para no detener los procesos importantes).

También podemos descolgarlo para hacer una llamada nosotros, este sería el caso de las interrupciones por programa que son originadas por la propia cpu desde el programa principal (con la instrucción BRK).

Después de todo este bollo, veamos cómo podemos aprovechar la interrupción para intercalar nuestros propios programas en ellas; en el mapa de memoria de ordenador nos encontramos con unos vectores de salto indirecto qu han sido colocados intencionadamente en memoria RAM para permitir alterarlos e incluir nuestras propias rutinas, los vectores son:

Nombre	Dirección
CINV	$0314-$0315
Vector de interrupción IRQ	
CBINV	$0316-$0317
Vector int. por instrucción BRK	
NMINV	$0318-$0319
Vector de interrupción NMI	

Como ejemplo del modo de utilizarlos podemos hacer un programa que permita mantener un rótulo fijo en la línea superior de la pantalla sin ser afectado por el desplazamiento de la pantalla (scroll), para ello ponemos la línea de texto en la zona que no sea invadida por el basic, y cada IRQ la recolocamos en la línea superior.

Programa para mantener la línea superior en el C-64.
.,033C A2 27 LDX #$27; 40 caracteres por línea.

(Viene de pág. 65.)

```
5040 DATA-22,-1,-1,-1,-1,22,-1,-1,-1
,-1,-1,-1,22
5050 DATA1,1,1,1,1,1,22,22,22,22,22,
22,22,22,22,-1,-1,-1
5060 DATA-22,-22,-22,-22,-22,1,1,1,-
22,-22,-22,-22,-22,-22,1,1,1,1
5070 DATA22,1,1,1,1,1,1,-22,22,22,22
,22,22,22
5080 DATA-1,-1,-1,-1,-1,-1,-1,-1,-1,
-1,-1,-1,-1,-1,-1,-1,22,22,22,22,22
5090 DATA1,1,1,1,1,1,-22,-22,-22,-22
,-22,1,1,1,1,22,22,22,22,22
```

```
5100 DATA1,1,1,1,1,1,-22,-22,-22,-22
,-22,-1,-1,-1,22,22,22,22,-1,-1,-1
5110 DATA-22,-22,-22,-22,-22,-22,-22
,-22,-22,-22,-1,-1,-1,-1
5120 DATA22,22,-1,-1,-1,22,22,22,22,
-1,-1,-1,22,22,22,22,1,1,1,-22,-22,-
22,-22
5130 DATA1,1,1,1,1,1,1,1,1,1,-22,-22
,-22,-22,-22,-22,22,-1,-1,-1,-22
5140 DATA-1,-1,-1,-1,22,-1,-1,-1,-22
,22,-1,-1,-1,22,22,22,22,22
5150 DATA1,1,1,22,22,22,-22,-22,-22,
-22,-22,-22,-22,1,1,1,-22,-22,9
```



EXCURSION POR LA MICROPROGRAMACION

```
., 033E BD 00 04 LDA $0400,X; línea superior.
., 0341 9D 74 03 STA $0374,X; la guarda.
., 0344 CA DEX
., 0345 10 F7 BPL $033E
., 0347 78 SEI
., 0348 AE 14 03 LDX $0314; guarda el vector.
., 034B AC 15 03 LDY $0315; de IRQ original.
., 034E 8E 72 03 STX $0372
., 0351 8C 73 03 STY $0373
., 0354 A2 60 LDX #$60; pone el nuevo
., 0356 A0 03 LDY #$03; vector
., 0358 8E 14 03 STX $0314
., 035B 8C 15 03 STY $0315
., 035E 58 CLI
., 035F 60 RTS
., 0360 A2 27 LDX #$27; vuelve a escribir
., 0362 AD 86 02 LDA $0286; los 40 caracteres
., 0365 9D 00 D8 STA $D800,X; en la pantalla
., 0368 BD 74 03 LDA $0374,X; con el color
., 036B 9D 00 04 STA $0400,X; actual del
., 036E CA DEX; cursor
., 036F 10 F1 BPL $0362
., 0371 4C 31 EA JMP $EA31; rutina normal.
```

Programa para el VIC-20 sin ampliación.

```
., 033C A2 15 LDX #$15
., 033E DB 00 1E LDA $1E00,X
., 0341 9D 74 03 STA $0374,X
., 0344 CA DEX
., 0345 10 F7 BPL $033E
., 0347 78 SEI
., 0348 AE 14 03 LDX $0314
., 034B AC 15 03 LDY $0315
., 034E 8E 72 03 STX $0372
., 0351 8C 73 03 STY $0373
., 0354 A2 60 LDX #$60
., 0356 A0 03 LDY #$03
., 0358 8E 14 03 STX $0314
., 035B 8C 15 03 STY $0315
., 035E 58 CLI
., 035F 60 RTS
., 0360 A2 15 LDX #$15
., 0362 AD 86 02 LDA $0286
., 0365 9D 00 96 STA $9600,X
., 0368 BD 74 03 LDA $0374,X
., 036B 9D 00 1E STA $1E00,X
., 036E CA DEX
., 036F 10 F1 BPL $0362
., 0371 4C BF EA JMP $EABF
```

Si el VIC esta ampliado, se debe cambiar $1E00 por la dirección donde se encuentra la memoria de pantalla ($1000) y el $9600 por $9400 que es donde se coloca la memoria de color.

Los dos programas deben ser tecleados con ayuda de un monitor de lenguaje máquina o ensamblador como ya expliqué en otro número, y es mejor guardarlo antes de utilizarlo, si lo hacéis en cinta es preferible hacer una rutina en basic con las líneas DATA correspondientes y el lazo FOR/NEXT con READ y POKE, ya que utiliza el buffer de cassette, y al intentar salvarlo se borra su contenido.

> **Esta rutina puede ser útil para algún programa en el que deseéis mantener el título en la pantalla, también podéis colocarla en otra dirección que no sea el buffer de cassette.**

Para ejecutar el programa es necesario teclear SYS 828, y la línea superior de la pantalla quedará "congelada", para quitarla pulsar STOP y RESTORE. Si se quiere hacer desde el modo de programa se puede introducir con la línea:

PRINT"[CLR] MENSAJE QUE QUEREMOS": SYS 828

Una observación: No teclear dos veces seguidas el SYS828 o congelaréis todo el ordenador al meterlo en un lazo cuando se produzca la primera IRQ, también podéis modificar la rutina para que esto no ocurra simplemente comprobando si el dato #$60 ha sido colocado anteriormente en la posición $0314 o aún no.

Esta rutina puede ser útil para algún programa en el que deseéis mantener el título en pantalla, también podéis colocarla en otra dirección que no sea el buffer de cassette si lo queréis, pero recordar cambiar los valores del vector de acuerdo con la posición que ocupe la rutina así como las instrucciones que utilizan direccionamiento absoluto.

Buena suerte con las interrupciones y no hagáis pruebas con programas antes de guardarlos en cinta, las sorpresas suelen ser desagradables y ni el OLD podrá ayudaros a recuperar todos los programas, pero la única manera de aprender es experimentando.

Las interrupciones son difíciles de manejar, pero merece la pena saber utilizarlas, una aplicación que me parece interesante es utilizarlas para controlar un buffer de impresora (SPOOLER), con el que podemos enviar ocho mil caracteres al buffer y el ordenador sigue enviándolos a la impresora de uno en uno sin detenernos un solo segundo mientras la impresora tarda un buen rato en escribirlos, en lugar de esperar como lo hacemos normalmente hasta que la impresora termina. Todo esto se podría hacer con el C-64 sin gastar ni un solo byte de la memoria de programas basic, utilizando la técnica de las interrupciones y los KBytes de RAM que están ocultos bajo la ROM, pero tranquilos, todavía no lo he hecho, es sólo una idea que está en fase de estudio... tardaré semanas o meses en sacar un rato para preparar el programa que lo haga... no obstante, aquí tenéis la idea si alguien quiere intentarlo... suerte.

MARKETCLUB

Servicio gratuito para nuestros lectores particulares.
De no especificar duración, los anuncios serán publicados durante 1 mes.
Debido a ciertos abusos que nos han sido comunicados, los anuncios gratuitos de Market club solamente serán publicados si vienen con nombre, apellidos y dirección completa. No se admitirán apartados postales.
Lamentamos que todos tengan que sufrir esta medida por culpa de algunos "*listos*"
Adicionalmente es recomendable que enviéis el número de teléfono por si hubiese alguna pega, si no deseáis que lo publiquemos podéis decirlo en la carta.

MERCADILLO

• Intercambio programas de cualquier tipo para CBM-64 tanto en disco como en cassette. Desearía contactar también con algún poseedor de compilador Pascal. O con personas interesadas en conseguirlo. Luis Paris García. Ctra. Vic. 119-121. 5º 2.ª Manresa - Barcelona. Teléf. 8730829 ***(Ref. M-139).***

• Deseo comprar la unidad disco VC-1541, así como los diskettes que me pudieran proporcionar quien vendiese este periférico, pero eso sí, en perfectas condiciones de funcionamiento. Las ofertas me las pueden enviar a la siguiente dirección. José M.ª Lozano Gallardo. Avda. de los Deportes, 42, 9º B. Móstoles. Madrid. ***(Ref. M-140).***

• Vendo VIC-20 + introducción al lenguaje de programación Basic I y II con cintas + cartuchos super Expander con 3K + Cartucho ajedrez "Sargón II Chess" por 45.000 pesetas. Llamar tarde. Teléf.: (967) 298282. Preguntar por Ernesto. ***(Ref. M-141).***

• Vendo VIC-20 comprado en mayo de 1983 con las 4 cintas + ampliación por 25.000 pesetas. Teléf.: (93) 8513636. Preguntar por Ramón. ***(Ref. M-142).***

• Vendo VIC-20 nuevo a estrenar a buen precio, llamar al teléfono: 8995153. Josep M.ª Carbonell Casanovas. C/Nov. 48. S, Pere de Riudevilles. Barcelona. ***(Ref. M-143).***

• Vendo programador universal de memorias EPROM para VIC-20 y CBM, graba las memorias siguientes, 2516-2716-2532-2564-2764-27128 y futuras. Félix Portabella Padro. C/Forn de Sta. Lucia, 1, 2º. Manresa. ***(Ref. M-144).***

• Vendo VIC-20 prácticamente nuevo (comprado en enero de 1984), 1 curso de Basic, cartuchos de juegos, juegos en cassette, libros y cassette por 39.000 ptas. Carlos Guardia. C/Córcega, 110, 1º 2.ª 08029. Teléf. 2397710. ***(Ref. M-145).***

• Vendo dos libros para el VIC-20: GAMES FOR YOUR VIC-20. 123 páginas (850). EXPLORING ADVENTURE GAMES ON THE VIC-20. 243 páginas (1.850). Andrés Portas Afenjar. Ríos Rosas, 5, 4ºD 29007 Málaga. ***(Ref. M-146).***

• Vendo VIC-20 en perfecto estado + Datasette C2N + Manual del Usuario + 1.ª parte del curso introducción al Basic + Cartucho (Omega Race) + cinta de indescomp (scramble) + Joystick anatómico de espectravideo + 12 revistas + cintas con juegos y programas de utilidad y vírgenes, todo ello por 31.900 ptas. Fco. Javier. Batanero del Amo. C/República Argentina, n. 62, 2º 4.ª Cornellá (Barcelona). Teléf.: 3763140 Prefijo 93. ***(Ref. M-147).***

• Intercambio juegos en cassette para C-64. Teléf. (93) 8032179. Bernat Planas Vives, Passeig Mn. Cinto Verdaguer 100 Atc. Igualada (Barna). ***(Ref. M-148).***

• Intercambio programas de juegos y aplicaciones para ORIC-1 Roger Noguera. C/Llessamins 10, Igualada (Barcelona). ***(Ref. M-149).***

• ¡Atención distribuidores! Deseo recibir información o catálogo sobre programas de juegos en cassette. A ser posible de venta en la provincia de Barcelona. Bernat Planas. Passeig Verdaguer, 100. Igualada (Barcelona). ***(Ref. M-150).***

• Vendo impresora nueva para C-64, 10 tipos de letra, gráficos, caracteres programables, etc. Luis Sánchez Pérez. Apartado nº 176. 45080 Toledo. Teléf.: 221494. ***(Ref. M-151).***

• Vendo Commodore-64 en perfecto estado (menos de un año de uso) junto con un gran número de programas: Interesados escribir a José Francisco Aldana Montes, Beethoven 6, 6º D 29004 Málaga o llamar al 317696 después de las 10. ***(Ref. M-152).***

• Urge vender Commodore 64, cassette, guía del programador, juegos, regalaría revistas, libros. Precio a convenir, también vendo por separado. Antonio Cebrián Martínez. C/Francisco Pizarro, 73. 02004 Albacete. Teléf.: (967) 234108 de 2 a 3,30 ***(Ref. M-153).***

• Vendo VIC-20, Cassette C2N, Cartucho de memoria de 16K, Super Expander, todo esto por 39.000 ptas. Los interesados llamar al (93) 699 58 80 y preguntar por José Navas. Juan José Navas Fuentes. C/Juan Ramón Jiménez, nº 26, 6º 3.ª de Rubí (Barcelona). ***(Ref. M-154).***

• Vendo monitor Commodore de Color, nuevo con garantía 60.000 ptas. Javier Taudes C/Corazón de María, 55. Madrid. Teléf. (91) 4167385 ***(Ref. M-155)***

• Intercambio programas para el Commodore 64 en cinta y disco, utilitarios y juegos. José M. Suárez., Rue J. Jeanfils nº 6. Soumagne. Belgique. ***(Ref. M-156).***

• Vendo Commodore Pet 2001 con interface para RTTY-CW, y programas por 50.000 ptas. Teléfono (942) 223985 de Santander. José Manuel ***(Ref. M-157).***

• Atención VICiosos: 1º Vendo Módulo de expansión de 16 K (para VIC-20). Intercambiaría juegos en cassette. Llamar o escribir a: Daniel Garriga. C.N. II, Waldem-7, 2-51. Sant Just Desvern (Barcelona). Tel.: (93) 3720243 ***(Ref. M-158).***

• Vendo Guía del Usuario para VIC-20 (1.500 ptas.) Guía de Referencia del Programador (1.500 ptas.) las dos partes del Curso Basic (1.700 cada uno; incluye cintas) y un cartucho de Ajedrez por 2.000 ptas. Todo por sólo 7.000 ptas. Salvador Tudela Edo. Gr. Carlos III, 130 "A". Barcelona 08034 - Teléf.: (93) 2041823 ***(Ref. M-159).***

• Soy usuario de un C-64. Desearía intercambio de información y programas de todo tipo (en cassette). Interesados dirigirse a: Marcel Planagumá. Pge. Mir i Borrell, 1-1º 2.ª Mataró (Barcelona). Tel. (93) 7985127 ***(Ref. M-160).***

• Vendo C-64 con 6 meses de uso, cassette Commodore, algunos juegos y el libro "60 programas para C-64" en inglés. Todo por 76.000 pts. Dirigirse a Antonio Torralba. C/Cayentana del Toro, 46, 11 K. Cádiz. Teléf.: (956) 27 88 70. ***(Ref. M-161).***

• Intercambio información y programas de todo tipo, en cassette, para C-64. M.ª Pilar Cabrerizo, calle José M.ª Durán, 51. 35010 Las Palmas. ***(Ref. M-162).***

• Vendo VIC-20, interface para conectarlo a cualquier cassette, curso de introducción al Basic (1ª parte), Guía del Usuario VIC-20, bastantes programas y muchas revistas COMMODORE, todo nuevo y en perfecto estado por 20.000 ptas. ¡Sí, has leído bien!, ¡todo junto por tan sólo 20.000 ptas.! Llamar desde las 3 h. en adelante al teléfono (93) 334 22 37. O escribir a José Antonio García Cortés, C/. Pintor M. Castellón, 4, 3º-3.ª. L'Hospitalet (Barcelona). ***(Ref. M-163).***

• Vendo C-64 + cassette + 30 programas originales (no son copias) por el precio de 85.000 ptas. (precio en el mercado, el doble). Antonio García de la Iglesia. C/ Juan Madrazo, 17-2º. León. Teléf. (987) 24 67 30, y 2 joysticks (uno profesional). ***Ref. M-164).***

• Vendo para C-64 "Curso de Introducción al BASIC I", y cartucho juego "RADAR RETRACE", o mejor aún lo cambiaría por Joystick, por "Guía de referencia C-64", o por programas de gráficas A.R. Manuel Morera Antoñanzas. C/Santa Inés, 4-1º C. 50003 Zaragoza. ***(Ref. M-165).***

• Tengo un C-64. Estoy interesado en el cambio de cintas (a ser posible de utilidades, gestión, lenguajes), así como en las ofertas que me puedan hacer de Unidad de Disco o Impresora. Mis señas son: José Marsa Mallol. C/Prats y Roqué, 32 Entlo. 1º. 08027 Barcelona. ***(Ref. M-166).***

• Vendo VIC-20 con ampliación 16 K RAM, cartucho SUPEREXPANDER + 3 K, cassette C2N, lote de cintas con juegos y utilidades, libros y manuales en inglés, colección Club Commodore, revistas inglesas y españolas. Todo por 45.000. Preguntar por Sindin en el teléfono 747 59 11 (mañana) de Madrid. ***(Ref. M-167).***

• Vendo VIC-20 con Superexpander 3 K, cassette, libros y cintas: 35.000 ptas. José Luis Manzano Linares. C/Martínez Izquierdo, 72. 28008 Madrid. Teléf. (91) 246 99 47. ***(Ref. M-168).***

• Vendo VIC-20 comprado en enero de este año, con ampliación de 16 K y datassete, adjunto varios programas comerciales y libros, además del joystick de Commodore. Todo por 29.000. Llamar al teléfono 31 01 17 (976). Zaragoza. ***(Ref. M-169).***

• Vendo 2 programas en cinta de ABC SOFT el "RENAISSANCE" (Otelo), este es un juego (en el 8 la máquina tarda de 10 a 15 minutos en tirar); 6 comandos de ayuda al jugador como marcha atrás, cambio de nivel, modificación del tablero, etc. También tiene la posibilidad de grabar/recuperar partidas. Se puede usar joystick o teclado. Instrucciones en Español; y el "MISSILE COMMAND", que es el juego de las salas recreativas en el que una base ha de proteger sus unidades del ataque de unos rayos extraterrestres. Se juega con joystick. Los 2 juegos por 4.000 ptas., por separado cada uno 2.500. Los 2 juegos son originales (no son copias). Antonio Mota Forta. Zaragoza, 10. Almería. Teléf. (951) 23 55 54. ***(Ref. M-170).***

• Vendo accesorios VIC-20, superexpander cassette, programas originales, las 2 partes del curso, guía de referencia, joystick e impresora GP 80. Teléf. 450 15 73. Vicente García Rudilla. C/ Bravo Murillo, 211-8º A. Madrid. ***(Ref. M-171).***

• Vendo VIC-20 comprado en el 84, con el datasette, ampliaciones de 3 K y de 16 K, joystick, 6 programas, curso de Basic en cassette, libros y muchos programas grabados. Todo en perfecto estado y casi sin usar por 42.000 ptas. Jesús Marcos Jamariz Ripalda. Teléf.: 254 10 52 de Barcelona. ***(Ref. M-172).***

• Vendo joystick para VIC-20 por 2.500 ptas., cinta nightmarepark por 1.200 ptas., cartridge vic avenger por 3.300 ptas. y el libro: "Acceso rápido al VIC 20" por 1.200 ptas. También vendo video-juegos Philips G-7000 por 10.000 ptas. y varios cartuchos de juegos para el mismo. Llamar en horas de comida a José Ramón de Diego Rodríguez. Teléf.: (987) 41 20 21. ***(Ref. M-173).***

CLUBS

• Club de amigos. Foto Estudio 2. Plaza de Sombrereros, 2. **Palma de Mallorca.** Tel.: 21 31 62. ***(Ref. C-1)***

• Club de programación Alaiz. Tels.: 254 480-257 704. Pamplona. ***(Ref. C-4)***

• Club de Commodore de Albacete. Fernando Martínez Guerrero. La Roda, 39. 5º D. Albacete. ***(Ref. C-5)***

• Círculo de Durensan Vigués. Vigo. C/Venezuela, 48. Entrechan. Tels.: (986) 410 683/422 519. ***(Ref. C-6)***

• Asociación Manchega de Usuarios de CBM-64 (A.M.U. C-64). C/Blasco de Garay, 10. Albacete. ***(Ref. C-13).***

• Club MICRO-ESPLAI. Gran Vía, 682. Pral. (Entre Bruch y Gerona) de Barcelona. ***(Ref. C-14).***

• Interesados en la formación de un club de informática en Granada capital diríjanse por correo a C/ San Jerónimo, 23, 2º. 18001 Granada. Miguel A. Puerta Carrasco. ***(Ref. C-28).***

• Si queréis que hagamos un club "Vic-20" poneros en contacto con Antonio Recober. Teléf.: 32 20 46. Málaga. Se trata de intercambiar programas. Conocimientos y experiencia sin que nos cueste dinero. Si algún 64 quiere hacer el camino a nuestro lado. Podemos intentarlo. ***(Ref. C-29).***

• Desearía contactar con interesados en formar un club de usuarios del C-64 en Granada e intercambiar programas y experiencias. Antonio Toral Peinado. C/Puente nº 3. Illora (Granada). Tel.: 958-46 30 09 ***(Ref. C-30).***

• Para formación club nacional de CBM Series 2000 y 3000 exclusivamente, ponerse en contacto por carta con Artemio González Pérez. Fernando Arocena Quintero, 1. Teléf.: 22 42 44. 38009 Santa Cruz de Tenerife. *(Ref. C-31).*

• Me gustaría contactar con alguien de Motril para compartir experiencias y/o formar un club de C-64 (y VIC-20 si hace falta). Ya tengo local. Mi teléfono: 958-600 314 *(Ref. C-32).*

• Ha nacido el club "Brian & Chip" para usuarios del C-64 y Spectrum 16K. Crearemos una revista propia, pero necesitamos colaboraciones, programas e ideas. Intercambiamos programas listados o en cinta, escribir desde cualquier rincón de España, prometo contestar. Francisco Reig López, C/Marina, 261, 3º 5ª, 08025 Barcelona. *(Ref. C-33).*

• Agradecería la dirección o teléfono de algún socio de mi ciudad o alrededores, a fin de ponerme en contacto con él para ver la posibilidad de formar un club o similar. José Esteban Llusar Molés. Daoiz, 2-1º 2ª. Pto de Sagunto (Valencia). *(Ref. C-34)*

• Grupo de usuarios de Commodore en Tarragona, estamos intentando formar un club lo más sólido posible, no nos mueve ningún interés lucrativo, y estamos desarrollando un sinfín de actividades, encaminadas a tener un más amplio conocimiento de nuestro ordenador. Alfred Blasi. Tel.: (977) 547 307. Tarragona. *(Ref. C-35).*

• Deseo contactar con niños-as que deseen formar un club de ordenadores Commodore 64 y VIC-20 (puede ser otra marca), para intercambiar experiencias. C/Avda. del Norte nº 8 atc. 3ª Hospitalet (Barcelona). Manuel González Martínez. Tel.: 93-240 23 50. *(Ref. C-36).*

• Agradecería mucho a todo el que pudiera informarme sobre la creación de un club para VIC-20 en Málaga. *(Ref. C-37)*

• Estamos montando una actividad de informática en el Club Los Chopos con la idea de enseñar a chavales de BUP y últimos cursos de EGB a programar y formar un grupo de amigos para intercambiar programas y experiencias, sin ningún fin lucrativo. Javier Huguet. C/Antonio Menchaca, 19. Teléf.: 4693461. Neguri. *(Ref. C-38).*

• Me gustaría contactar con interesados en la confección de programas de astronomía. Ordenador VIC-20. Interesados, escribir a: Apartado 2309. 15080 La Coruña. *(Ref. C-39).*

N.R.: Para continuar saliendo en el futuro, envíanos Dirección.

• Los interesados en programas (VIC-20) relacionados con la frecuencia de 27 Mhz escriban a: Apartado Postal 471. 14080 Córdoba. *(Ref. C-40).*

• Estamos formando un Club de Usuarios en el centro de Madrid. Cualquier interesado que se ponga en contacto con nosotros en el teléfono 2766694, Lunes-Miércioles-Viernes de 18,00 a 20,00 h. Dirigiéndose a Rafael (EB4XD). (Dejar teléfono en el contestador). *(Ref. C-41).*

N.R.: Para continuar saliendo en el futuro, envíanos Dirección.

• Me gustaría contactar con usuarios de la zona de Mataró. Dirigirse a: Marcel Planagumá. Pge. Mir i Borrell, 1-1º 2ª. Mataró (Barcelona). Tel. (93) 7985127. *(Ref. C-42).*

• Desearía información sobre clubs en existencia y gente que quiera formar uno o intercambiar programas y experiencias preferentemente de Pontevedra. (Sólamente usuarios de Commodore 64). Carlos Parada Gandos. C/Padre Fernando Olmedo, 4-4º B. 36002 Pontevedra. Teléf.: (986) 85 69 21. *(Ref. C-43).*



• Profesor de inglés, trabajando en un instituto y licenciado en Historia, daría clases particulares de inglés. Dirigirse a: Michael Bruggeman. Teléf.: (91) 2271143. *(Ref. BD-1).*

PARA LLEGAR MAS LEJOS...

SPECTRAVIDEO™



MKII







MICROSOFT BASIC ★ FORTRAN 80 ★ Microsoft Logo ★ **COLECO cartuchos**
CP/M ★ Cobol 80 ★ MACROENSAMBLADOR Z80 ★ **Ensamblador CP/M**

- **MSX. Es el sistema escogido por 16 fabricantes japoneses de ordenadores** (SONY, PANASONIC, PIONER, SANYO, SPECTRAVIDEO, etc.) **siendo compatibles todos los programas creados por cada uno de los fabricantes.**

CARACTERISTICAS GENERALES	SPECTRAVIDEO 328	SPECTRAVIDEO 318	APPLE II e	COMMODORE 64	ATARI 600 xL	BBC MODELO B	DRAGON 32	ORIC ATMOS
PROCESADOR	Z80 A	Z80 A	6502	6510	6502	6502	6809 E	6502
RELOJ EN MHz	4 MHz	4 MHz	1 MHz	1 MHz	1.8 MHz	2 MHz	1 MHz	1 MHz
MEMORIA STANDARD EN RAM	80 K	32 K	64 K	64 K	16 K	32 K	32 K	48 K
AMPLIABLE HASTA	144 K	144 K	–	–	64 K	–	64 K	64 K
MEMORIA STANDARD EN ROM	32 K	32 K	16 K	20 K	24 K	16 K	16 K	16 K
AMPLIABLE HASTA	96 K	96 K	–	–	–	–	–	–
MICROSOFT BASIC EXTENDIDO	SI	SI	–	–	–	NO	SI	SI
NUMERO DE TECLAS	87	51	62	66	57	73	53	57
TECLADO CON GRAFICOS PREDEFINIDOS	SI	SI	–	SI	SI	SI	NO	NO
TECLAS DE FUNCION PROGRAMABLES	10	10	–	8	4	10	NO	NO
MANDO JOYSTICK INCORPORADO	NO	SI	–	NO	NO	NO	NO	NO
COLORES	16	16	16	16	16	16	9	8
SPRITES	32	32	–	8	8	–	?	–
RESOLUCION (Puntos de pantalla)	256×192	256×192	280×192	320×200	320×192	256×640	256×192	240×200
TEXTO EN PANTALLA	40×24	40×24	40×24	40×25	40×24	40×32	32×16	40×28
CANALES DE SONIDO	3	3	1	3	4	1	3	3
OCTAVAS POR CANAL	8	8	4	9	?	3	5	8
A.D.S.R. ENVOLVENTE	SI	SI	NO	SI	NO	SI	NO	SI
CAPACIDAD UNIDAD DE DISCO	256 K	256 K	140 K	170 K	127 K	100 K	?	?
CENTRONICS	SI (opcional)	SI (opcional)	SI (opcional)	–	SI (opcional)	SI (opcional)	SI	SI
RS232	SI (opcional)	SI (opcional)	SI (opcional)	–	SI (opcional)	SI (opcional)	–	–
80 COLUMNAS	SI (opcional)	SI (opcional)	SI (opcional)	SI (opcional)	SI (opcional)	–	–	–
PRECIO P.V.P.	76.000	49.900	166.618	79.900	58.500	140.000	67.800	59.500·

(TM)

- MSX y MICROSOFT son marcas registradas por MICROSOFT CORPORATION.
- CPM es una marca registrada por DIGITAL RESEARCH INC.
- COLECO VISION es una marca registrada por COLECO INDUSTRIES INC.



HARDWARE
DISTRIBUIDOR OFICIAL DE SPECTRAVIDEO INTERNATIONAL LTD
PASEO DE LA CASTELLANA, 179 - MADRID-16
TELEFONO (91) 279 31 05

DELEGACION EN CATALUÑA: ACE, S.A.
Tarragona, 100 - Barcelona-15.
Teléfono (93) 325 10 58

YA DISPONIBLE EN:



... Y EN TODAS LAS TIENDAS ESPECIALIZADAS.



COMMODORE 16
LA EMOCION DE EMPEZAR

Iniciarse en el mundo de los ordenadores personales con un COMMODORE 16 es sumar, a la emoción de empezar, la emoción del futuro.

Porque es un ordenador de fácil manejo y programación, pero con prestaciones que sólo se encuentran en ordenadores de costo mucho más elevado.

Porque es un ordenador pequeño, pero con la mayor cantidad de software y periféricos, que multiplican sus posibilidades futuras.

Porque es el ordenador ideal para empezar y perfecto para seguir.

PRINCIPALES CARACTERISTICAS	
— 16 K. — COMANDOS DE ALTO NIVEL PARA GESTION DE COLOR, SONIDO Y GRAFICOS DE ALTA RESOLUCION. — TECLADO PROFESIONAL.	— 40 COLUMNAS × 25 LINEAS. — 121 COLORES. — GRAFICOS EN ALTA RESOLUCION. — 2 GENERADORES DE TONO. — AMPLIA GAMA DE PERIFERICOS.

commodore 16

LA EMOCION DEL FUTURO





MICROELECTRONICA Y CONTROL
c/ Valencia, 49-53. 08015 Barcelona. c/ Princesa, 47, 3.° G. 28008 Madrid.
UNICO REPRESENTANTE DE COMMODORE EN ESPAÑA

SLOGAN